

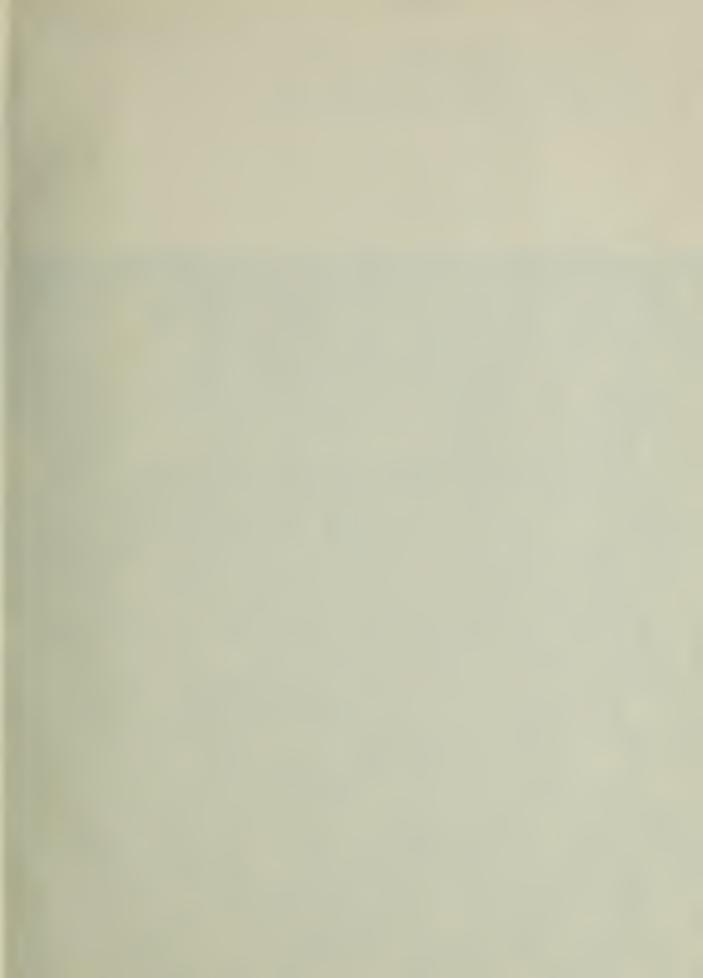
The person charging this material is responsible for its return to the library from which it was withdrawn on or before the **Latest Date** stamped below.

Theft, mutilation, and underlining of books are reasons for disciplinary action and may result in dismissal from the University.

UNIVERSITY OF ILLINOIS LIBRARY AT URBANA-CHAMPAIGN

CT 1 5 RECU





Digitized by the Internet Archive in 2013

Tlah UIUCDCS-R-75-698 No. 698 Cop 2

PROGRAM MANUAL:

NOR NETWORK TRANSDUCTION BASED ON CONNECTABLE AND DISCONNECTABLE CONDITIONS

(Reference Manual of NOR Network Transduction Programs NETTRA-Gl and NETTRA-G2)

bу

February 1975

J.N. Culliney

THE LIBRARY OF THE

APA 1 1975

HAMMERGITY OF HUMBIS



DEPARTMENT OF COMPUTER SCIENCE
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN · URBANA, ILLINOIS



# UIUCDCS-R-75-698

PROGRAM MANUAL:

NOR NETWORK TRANSDUCTION BASED ON CONNECTABLE

AND DISCONNECTABLE CONDITIONS

(Reference Manual of NOR Network Transduction

Programs NETTRA-Gl and NETTRA-G2)

by J.N. Culliney

February 1975

Department of Computer Science University of Illinois at Urbana-Champaign Urbana, Illinois 61801

This work was supported in part by the National Science Foundation under Grant No. GJ-40221.



510.84 Illor no.698-702 cop2

## ABSTRACT

This paper explains the operation and usage of two FORTRAN computer programs, NETTRA-Gl and NETTRA-G2, developed for NOR-network transduction (transformation and reduction).

Existing (non-optimal) NOR-gate networks and their required output functions are given to the programs as input. The programs, in general, add, change, and/or delete connections in the network in an effort to reduce the cost of the network (defined in terms of numbers of gates and connections) as much as possible. Gates are examined individually; their input connections and potential input connections are evaluated under certain conditions of connectability and disconnectability in order to effect the changes in network configuration and thus reduce network cost.

These programs are only two out of a whole system of programs, designated by the name "NETTRA" (for NETwork TRAnsduction), which implement different NOR-network transduction procedures.

The theoretical basis for the algorithms implemented by NETTRA-Gl and -G2 is detailed in earlier reports ([1] and [2]).



## ACKNOWLEDGMENT

The author is greatly appreciative of Prof. S. Muroga for his discussions and guidance relating to the preparation of this paper, and also, for his careful reading and valuable suggestions for the improvement of the original manuscript. The author is also indebted to H.C. Lai and Y. Kambayashi upon whose related research much of the work reported herein depends.

This work was supported in part by the National Science Foundation under Grant No. GJ-40221.



# TABLE OF CONTENTS

1.	INTRODUCTION
2.	A GENERAL NETWORK TRANSDUCTION PROCEDURE TO REMOVE NON-SPECIFIC
	GATES
3.	A SPECIALIZED NETWORK TRANSDUCTION PROCEDURE TO REMOVE A SPECIFIC
	GATE
4.	MAJOR FUNCTIONS OF COMMON SUBROUTINES
5•	INPUT DATA SETUP
	5.1 Input Data Card Format
	5.2 Restrictions on Problem Size
	5.3 Examples of Input Data Setup
REF!	ERENCES
APP	ENDIX: PROGRAM LISTINGS



## 1. INTRODUCTION

This manual is intended to instruct the reader in the use of the FORTRAN programs 'NETTRA-Gl' and 'NETTRA-G2'. These programs realize the algorithms described in detail in [1], and this manual will assume a knowledge of those definitions and algorithms in [1] as well as a knowledge of the general description of network transduction procedures presented in [2].

NETTRA-Gl and -G2 are only two out of a whole system of programs developed at the University of Illinois by the research group led by Prof. S. Muroga. The generic name 'NETTRA' (for NETwork TRAnsduction) designates the whole collection of programs comprising the system. All of the programs in the NETTRA system either transform or assist in transforming networks of interconnected NOR gates realizing various functions of their respective sets of input variables. By these transformations, a large, non-optimal network of NOR gates realizing one or more various functions can often be reduced to a smaller, less expensive (in terms of the number of required gates and interconnections, for example), near-optimal network realizing the same functions(s). In general, such a transformation could involve a complete reorganization of the network: the addition and/or deletion of gates; the addition and/or deletion of connections among gates; and/or the substitution of certain connections for various others. The procedures realized by NETTRA-Gl and NETTRA-G2 can accomplish any of these changes, with the exception of adding gates to the network.

The procedures realized in the programs NETTRA-G1 and -G2 are more complex than those appearing in [3] and require more computer time to execute. However, they are more powerful also, and they can often reduce

a network when it is impossible to do so by those other procedures.

The programs, NETTRA-PGl, -Pl, and -P2 described in [3], though, are more efficient than NETTRA-Gl and -G2 when first applied to large, far-from-optimal networks. NETTRA-Gl and -G2 are most useful when applied to more nearly optimal networks where it is fairly difficult to achieve a further reduction of the network.

As can be seen later, NETTRA-Gl and -G2 are quite similar programs, sharing the same major subroutine in fact; but they realize significantly different transformation procedures.

The next two sections, Sections 2 and 3, discuss the two programs in greater detail and present some examples of the effectiveness of their transformations. This is followed, in Section 4, by a description of the functions of the subroutines which support the subroutines actually realizing the procedures. Section 5 outlines the preparation of input for the two programs. Finally, in the appendix, a complete listing of each of the FORTRAN programs, NETTRA-Gl and NETTRA-G2 is given.

## 2. A GENERAL NETWORK TRANSDUCTION PROCEDURE TO REMOVE NON-SPECIFIC GATES

This and the following section will discuss, respectively, the NOR-network "transduction" (transformation and reduction) procedures realized by the FORTRAN programs designated NETTRA-Gl and NETTRA-G2. These programs realize procedures which are strictly network reduction procedures. In other words, when they are applied in an attempt to transform a network, the cost of the network will be either reduced or unchanged - it will never be increased.

The input to either of these programs is a description of a particular NOR network under consideration. This description (explained in detail in Section 5) consists of a set of various network parameters. The output of both programs is a description of the "transformed" network (if a transformation was possible).

As opposed to the procedure to be discussed in Section 3, the procedure discussed in this section does not attempt to remove specific gates from a network. A calculation is made, during which, usually, the network connection pattern is altered and unnecessary gates are recognized and removed from the network. Due to certain "orderings" which are needed in the procedure to perform selection decisions, there is a definite "preference" to remove certain gates from the network rather than certain others; but the procedure definitely does not focus its power on attempting the removal of a specific gate.

As a by-product of the calculation, compatible sets of permissible functions are generated. This information is necessary for the further

Refer to [1] for definitions of unfamiliar terminology.

application described in Section 2.2 of [3].

The entire NETTRA-Gl program requires 144 K bytes of core storage, about 59 K being occupied by the actual program instructions and about 85 K by the stored data (compiled by FORTRAN H (OPT 2) compiler).

The following subroutines, written in FORTRAN IV for the IBM 360/75, consititute the program NETTRA-Gl: MAIN, PROCII, CONCCO, ELANDO, MINI2, RNONES, SUBNET, and OUTPUT. Two system-supplied timing routines, STIMEZ and KTIMEZ are also assumed to be available, but if they are not, their use can be omitted from the program, or another suitable timing routine substituted, without harming the procedure itself. The functions of the support subroutines, MAIN, CONCCO, ELANDO, MINI2, RNONES, SUBNET, and OUTPUT, will be discussed in Section 4.

The general organization of the program NETTRA-Gl is shown in Fig. 2.1. An arrow from block i to block j represents the fact that the subroutine represented by block i calls the subroutine represented by block j.

# 2.1 Flowchart of the Subroutine Realizing the Procedure

Although NETTRA-Gl is composed of eight subroutines, the logic realizing the transduction procedure is essentially embodied in just one subroutine: PROCII (for PROCedure II, a name assigned during the development of the program). The following discussion of PROCII will assume a general knowledge of the information contained in [1].

Explanations of the purposes of the variables and arrays appearing in the subroutine can be found in the program listing of PROCII in the appendix. It is, however, convenient to define some of the variables at this point in order to discuss the flowchart of PROCII which appears in Figs. 2.1.1 and 2.1.2.

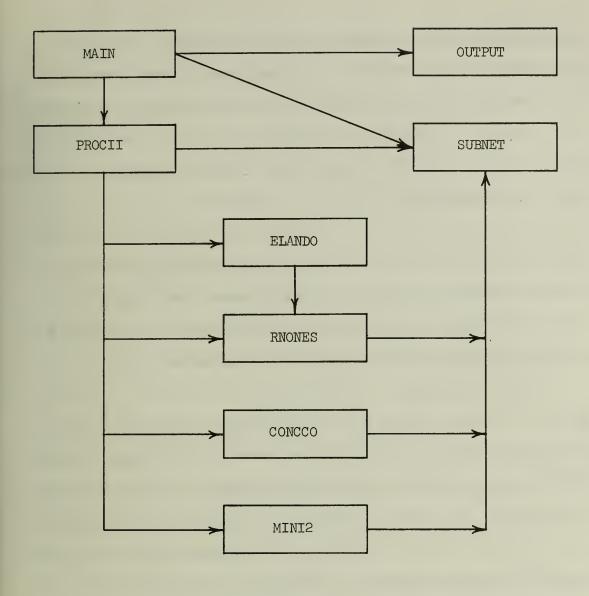


Fig. 2.1 General organization of the program NETTRA-Gl.

N is the number of external variables, n, if only uncomplemented variables are allowed as inputs. If both complemented and uncomplemented are available (i.e., n variables and their n complements) then N is equal to 2n. Note that this is strictly the representation <u>internal</u> to the program; for input-output purposes (as described in Section 5) N and n are always equal.

R is the number of gates specified by the input data to the program. It includes all gates declared to be present by the input data, even though some of them may be isolated (i.e., not connected to other gates in the network). Internally, the program represents the gates 1, 2, ..., R by the labels N + 1, N + 2, ..., N + R. (External variables are labeled 1, 2, ..., N internally.)

 $\overline{NR}$  is equal to the sum N + R. It is often convenient to treat both external variables and gates in a similar manner. External variables being labeled 1, 2, ..., N and gates being labeled N + 1, ..., N + R (internally), the number N + R is frequently required.

GORDER is an array containing a certain ordering of the various gate and external variable labels. In other words, in the locations GORDER(1), GORDER(2), ..., GORDER(NR) are stored the numbers 1, 2, ..., NR in a certain order. The ordering represented by the array GORDER satisfies the following criterion: for every gate or external variable, i, which feeds another gate, j, gate j precedes gate i in the ordering (for this to be possible, the network is assumed to be loop-free).

GSMALL is a two-dimensional array used to store the intermediate and final calculated compatible sets. GSMALL entries are initialized to "don't-cares" at the beginning of the procedure; upon termination of the algorithm, the determined compatible sets can be read directly from GSMALL.

For simplicity, sometimes just the words "compatible sets" will be used to denote compatible sets of permissible functions.

GSMALL(i, j) contains (or rather, will contain by the end of the procedure) the j<sup>th</sup> component of the vector representing the compatible set of permissible functions for gate or external variable i.

In <u>block l</u> of the flowchart (Fig. 2.1.1), an ordering of gates and external variables is determined and stored in the array GORDER. Also, at approximately this point in the program, initial values are assigned to many of the arrays and variables which will be used later.

Block 2 simply initializes a counter, the variable GCOUNT, used in the program loop consisting of blocks 3, 4, 7, 8, and 9.

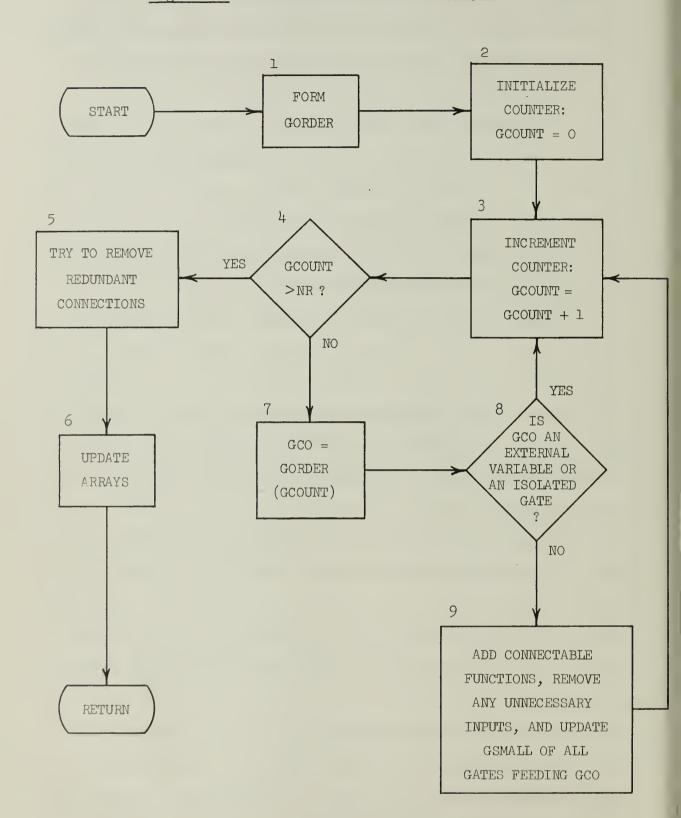
Block 3 increments the counter, GCOUNT. The program loop formed by blocks 3, 4, 7, 8, and 9 executes once for each value of GCOUNT: 1, 2, 3,

When GCOUNT is incremented beyond the value NR, it is detected in block 4. This is a sign that the algorithm has finished, having scanned every gate in the network, and the program enters block 5. Otherwise, the program proceeds to block 7.

In <u>block 5</u> the algorithm has essentially finished. A subroutine (MINI2) is called which quickly searches for and removes certain redundant connections which may still remain in the network (for example, certain new connections that might have been added unnecessarily in block 9). The removal of such connections does not cause a change in the output functions of the network.

At the end of every transformation there are certain "house-keeping" chores which must be performed: updating or restoring values in arrays and variables. This task is done in <u>block 6</u>. This is followed by a return to the calling subroutine (MAIN).

Fig. 2.1.1 Generalized flowchart of PROCII.



In <u>block 7</u>, the variable GCO is assigned the label (of a gate or external variable) contained in the (GCOUNT)<sup>th</sup> position of the ordering stored in the array GORDER. GCO becomes the name of the gate (or external variable) about to be examined by the program.

If GCO happens to be either an external variable or an isolated gate, no action needs to be taken. In such a case <u>block 8</u> sends control back to block 3. Otherwise, the program continues to block 9.

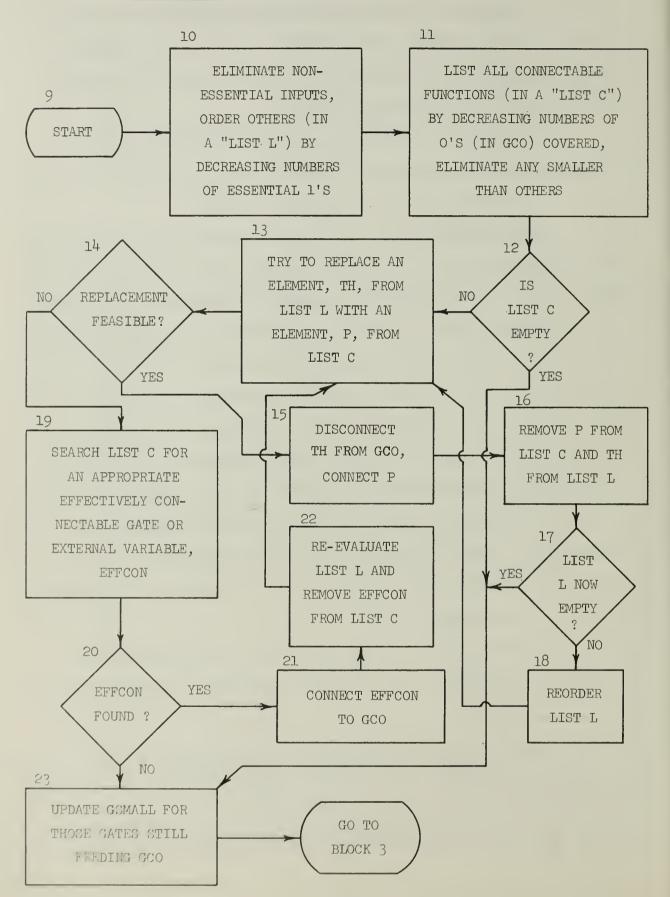
Block 9 performs many functions and is actually quite complex. So block 9 is detailed in Fig. 2.1.2 as consisting of sub-blocks 10 through 23. In these blocks the program focuses its attention on gate GCO and its inputs.

During the previous part of the calculation, the output requirement (i.e., the set of permissible functions<sup>†</sup>) of gate GCO (0, 1 or \*) has already been determined for each of the 2<sup>n</sup> possible combinations of external variable values. Whenever the required output (i.e., GSMALL component) of GCO is a 1 for a certain input vector, all of the immediate successors of GCO (i.e., the gates and/or external variables which feed GCO) must have an output of 0. Also, whenever the required output of GCO (i.e., GSMALL component) is a 0, at least one of its immediate successors must have an output of 1. If the required output of GCO is \* for some input vector, the outputs of its immediate successors are unrestricted.

In block 9, it is the positions of the 0's in the permissible function vector of GCO which are of foremost importance. Due to the nature of

A set of permissible functions for a gate I is represented by a  $2^n$ -dimensional vector [stored in (GSMALL(I, J), J=1, 2, ...,  $2^n$ )], ( $f^{(1)}$ , ...,  $f^{(2^n)}$ ), where  $f^{(j)} = f(x_1, x_2, \ldots, x_n)$  for  $j-1 = 2^{n-1}x_1 + 2^{n-2}x_2 + \ldots + x_n$ . Each coordinate is a 0, 1, or \*. If \*'s are used, then the vector represents the set of all functions which are obtained by assigning 0 or 1 to each \* in all possible ways.

Fig. 2.1.2 Detail of block 9 of PROCII flowchart.



NOR gates, a l appearing on any of the input lines to a gate will cause a O output of that gate. Such a l is called a <u>cover</u> of that O, and the O is said to be <u>covered</u> by that l. Although a O output may be covered by several l's (appearing on different input lines), only a single cover is actually required to guarantee the O output.

Block 10 actually represents two steps. The first step is the elimination of non-essential inputs of GCO.

Checking the immediate predecessors to GCO, the number of 1 covers for each 0 component of the permissible function vector is determined and stored. Some 0 components have only a single 1 cover. In such a case, the single 1 covering the 0 is called an <u>essential 1</u>. Any input line (connection) carrying an essential 1 for some input vector cannot be disconnected from GCO without causing the actual function of GCO to be outside the set of permissible functions for GCO. However, the input connections to GCO which do not carry essential 1's are redundant and are removed from the network in the first step of block 10. The removal of these redundant input connections must be done serially though since removing any input to a gate might cause new essential 1's to be created.

The remaining inputs to GCO all have essential l's. They are ordered in a list L, (realized by an array named "LISTL" in the program) such that the input represented by LISTL(i) has at least as many essential l's as the input represented by LISTL(i+1). This is the second step of block 10.

In block 11, first a search is made for all gates and external variables which are effectively connectable to GCO. (The original inputs to GCO are prohibited from being considered effectively connectable to GCO.)

If one of these effectively connectable functions, call it A, covers every O of GCO covered by another effectively connectable function, say B, then B is

<sup>†</sup> Refer to [1] for definitions of unfamiliar terminology.

eliminated from the collection of (effectively connectable) functions (i.e.,

A is greater than or equal to B in every component corresponding to a position
of a O in the permissible function vector of GCO). Those remaining effectively
connectable functions are ordered in a list C, (realized by an array named
"LISTC" in the program) such that the function represented by LISTC(i) covers
at least as many O's of (the permissible function vector of) GCO as the
function represented by LISTC(i+1).

If the program is unable to detect any effectively connectable functions (i.e., if list C is empty), control goes to block 23 and then to block 3. Otherwise the program proceeds to the major program loops in block 9, consisting of blocks 13 through 22. Block 12 tests for an empty list C.

Block 13 seeks an element, <sup>†</sup> P, from list C which can be directly substituted for an element, TH, from list L, such that the substitution would not cause the actual function of GCO to be outside its set of permissible functions. Furthermore, this TH and P are also selected in such a manner that TH is the first element in list L which can be replaced by an element in C and that P is the first element in list C which can replace that TH.

Block 14 tests if a feasible replacement has been found. If no replacement is possible, control passes to block 19. If, however, a suitable P and TH were chosen, blocks 15 through 18 perform the actual exchange of P for TH as an input of gate GCO.

First TH is disconnected from GCO in <u>block 15</u>. Also this block connects the new input, P, to GCO.

In this case, an "element" is actually the function realized by a gate or external variable.

Block 16 removes P from the list C of effectively connectable functions and TH from the list L of inputs to GCO. If list L becomes empty by the removal of TH (block 17 test), the program has replaced all of the original inputs to GCO by new ones, and the program moves to the next step of the procedure in block 23.

Otherwise the program goes to <u>block 18</u> where the elements of list L are reordered by decreasing numbers of essential l's. This is necessary since, by the addition of P to GCO, some previously essential l's may have become non-essential. From here, the program returns to block 13 to search for another replacement P.

Block 19 is reached when it is no longer possible to replace an element of list L with an element of list C as an input to GCO. Here, the program first searches for an element of list C which can cover at least one O of (the permissible function vector of) GCO which is currently covered by an essential 1 belonging to one of the original inputs to gate GCO. If such an input is found, it is assigned the label EFFCON and the program proceeds to block 21. If no EFFCON can be chosen, this implies that there can be no further replacements of original inputs to GCO by elements of list C. In this case, the program searches list C one last time looking for elements which cover at least one O of GCO which is currently covered only by the remaining original inputs to GCO (i.e., which is not covered by any of the newly connected functions). The group of elements satisfying this criterion are connected to GCO, and control goes to block 23.

Block 20 was discussed as part of block 19.

In block 21 the selected EFFCON is connected to GCO.

This connection requires the reordering of list L and the removal

of EFFCON from list C. This is done in <u>block 22</u>. The program then returns to block 13 to try again to find an element of list L which can be replaced by an element of list C. This may now be possible although it was impossible before the connection of EFFCON to GCO.

In <u>block 23</u> the covering assignments are made for the gates still feeding GCO. In other words, for each 0 component of the permissible function vector of GCO, one of the gates, GI (feeding GCO), producing a lacover for that 0 is selected. Gate GI is then <u>required</u> to produce a lacover for that 0 component, and this requirement is actually a restriction on the set of permissible functions for GI. This requirement is recorded (assuming the 0 in question appears as the jth component of the permissible function vector of GCO) by forcing the jth component of the permissible function vector of GI to be a lace, the value lace is stored in the location GSMALL (GI, j)). Although other lace, they are not "required" in the same sense as the lacever provided by the selected gate.

After all of the required 1 outputs of gates feeding GCO have been selected and stored (in GSMALL), the required 0 outputs of the immediate predecessors (actually, external variables feeding GCO can be, and are, ignored) are determined and stored by inserting 0's into GSMALL in the appropriate locations. It is an easy task to find the locations of these required 0's. If 1's appear in GSMALL (GCO,  $j_1$ ), GSMALL (GCO,  $j_2$ ), ..., GSMALL (GCO,  $j_k$ ) [i.e., in the  $j_1^{th}$ ,  $j_2^{th}$ , ...,  $j_k^{th}$  components of GCO's permissible function vector], then 0's must be required in GSMALL ( $p_1$ ,  $p_1$ ), ..., GSMALL ( $p_2$ ,  $p_1$ ), ..., GSMALL ( $p_2$ ,  $p_2$ ), ...;

If the 0 is found to be covered by an external variable, no 1 cover is selected.

GSMALL  $(p_{\ell}, j_{1}), \ldots, (p_{\ell}, j_{k}),$  where  $p_{1}, p_{2}, \ldots, p_{\ell}$  are the labels of the immediate predecessors of GCO.

The completion of block 23 also means the completion of block 9, and execution of the program moves into block 3 of Fig. 2.1.1.

## 2.2 Example for NETTRA-Gl

Fig. 2.2.1 shows partially the printout obtained from NETTRA-Gl for a typical example.

The original network produces a single output function and consists of 25 gates and 105 connections. Five independent, uncomplemented variables are available as inputs to the network. This information appears at the beginning of the output (Fig. 2.2.1 (a)).

This is followed by a complete truth table (b) showing the output of every gate in the original network for every possible input combination.

Note it is gate 1 which realizes the output function of the network.

Next appears a description of the configuration of the network (c). Each gate is listed along with the gates and/or external variables which are its inputs. The level numbers, also to be seen in (c), will be discussed in Section 5.3.

The truth table (note that the outputs for disconnected gates are shown as all 1's) and network configuration for the transformed network resulting from the action of NETTRA-Gl are shown in (d) and (e), respectively. The derived network consists of 11 gates and 38 connections. If NETTRA-Gl were applied to this new network, a third network of 10 gates and 36 connections would be obtained.

# Fig. 2.2.1 Printout for a Network Transformed by NETTRA-Gl.

\*\*\*\* 5 VAR., EXAMPLE

HEX=FF68AlF3

NUMBER OF VARIABLES = 5

NUMBER OF FUNCTIONS = 1

COST COEFFICIENT A = 100

B = 1

--- UNCOMPLEMENTED VARIABLES X ---

FUNCTION NO. 1.
11111111011010001010000111110011

ORIGINAL NETWORK COST = 25105

(a) Heading information and network parameters.

## TRUTH TABLE

```
X3 = 0 0 0 0 1 1 1 1 1 0 0 0 0 1 1 1 1 1 0 0 0 0 1 1 1 1 1 0 0 0 0 1 1 1 1 1
1 = 1 1 1 1 1 1 1 1 0 1 1 0 1 0 0 0 0 1 0 1 0 0 0 0 1 1 1 1 1 1 0 0 1 1
10 = 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
12 = 0 0 0 0 0 0 0 0 0 0 0 1 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0
24 = 1 1 0 0 0 0 0 0 1 1 0 0 0 0 0 0 1 1 0 0 0 0 0 0 1 1 0 0 0 0 0
```

```
GATE .. LEVEL
               FED BY
  1
        / 1/
               3 7 10 12 15 17 19 21 23 25
        / 3/
               X1 X2 X3 X4 X5
  2
        / 2/
              X1 X3 X4 X5 2
  3
  4
        / 3/
               X1 X2 X3
       / 3/
               X1 X3 X4
  5
  6
       / 3/
               X1 X3 X5
               X1 X3 4 5 6
  7
        / 2/
  8
        / 3/
               X1 X2 X5
        / 3/
               X1 X4 X5
  9
        / 2/
               X1 X5 6 8 9
 10
        / 3/
 11
               X1 X2
       / 2/
 12
               X1 5 8 9 11
        / 3/
               X1 X2 X3 X4
 13
 14
        / 3/
              X2 X3 X4 X5
        / 2/
              X2 X3 X4 13 14
 15
        / 3/
 16
               X2 X3 X5
        / 2/
               X2 X3 4 13 16
 17
 18
        / 3/
               X1 X2 X4
               X2 X4 14 18
 19
        / 2/
              X1 X2 X4 X5
        / 3/
 20
        / 2/
 21
              X2 X5 8 14 16 20
        / 3/
              X3 X4 X5
 22
        / 2/ X4 X5 9 20 22
 23
       / 3/ X3 X4
 24
        / 2/ X4 9 18 20 24
 25
```

<sup>(</sup>c) Configuration of original network.

## NETWORK DERIVED BY PROCII TIME ELAPSED = 144 CENTISECONDS

## TRUTH TABLE

```
X3 = 0 0 0 0 1 1 1 1 1 0 0 0 0 1 1 1 1 1 0 0 0 0 1 1 1 1 1 0 0 0 0 1 1 1 1 1
X^{\downarrow} = 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1 0 0 1 1
1 = 1 1 1 1 1 1 1 1 0 1 1 0 1 0 0 0 0 1 0 1 0 0 0 0 1 1 1 1 1 1 0 0 1 1
12 = 0 0 0 0 0 0 0 0 0 0 0 1 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
24 = 1 1 0 0 0 0 0 0 1 1 0 0 0 0 0 0 1 1 0 0 0 0 0 0 1 1 0 0 0 0 0
```

<sup>(</sup>d) Truth table for transformed network.

GATE	LEVEL			FED BY				
1		/	1/	3	12	17	21	25
2		/	1/					
3		/	2/	Xl	Х3	Х4	X5	11
4		/	1/					
5		/	1/					
6		/	1/					
7		/	1/					
8		/	1/					
9		/	3/	Xl	X4	X5		
10		/	1/					
11		/	3/	Xl	X2			
12		/	2/	Xl	9	11	16	24
13		/	1/					
14		/	1/					
15		/	1/					
16		/	3/	Х3	X5			
17		/	2/	X2	Х3	11	22	
18		/	1/					
19		/	1/					
20		/	1/					
21		•	2/	X2	X5	11	16	24
22		/	3/	X5				
23		/	1/					
24		/	3/	Х3	X4			
25		/	2/	X4	9	11	24	

<sup>\*</sup> A NETWORK DERIVED BY PROCII COST = 11038.

<sup>(</sup>e) Configuration of transformed network.

# 3. A SPECIALIZED NETWORK TRANSDUCTION PROCEDURE TO REMOVE A SPECIFIC GATE

The procedure about to be discussed here attempts to remove <u>specific</u> gates from a network, in contrast to the procedure just described. Actually, this procedure, realized by the program NETTRA-G2, consists of many applications of a smaller procedure which attempts (upon each application) to remove a specific gate from the network.

NETTRA-G2 consists of the following subroutines written in FORTRAN IV for the IBM 360/75: MAIN, PROCIV, PROCII, MINI2, CONCCO, ELANDO, RNONES, SUBNET, and OUTPUT. Two system-supplied timing routines, STIMEZ and KTIMEZ are also assumed to be available, but if they are not, their use can be omitted from the program, or another suitable timing routine substituted, without harming the procedure itself. The functions of the support subroutines MAIN, CONCCO, ELANDO, MINI2, RNONES, SUBNET, and OUTPUT will be discussed later in Section 4.

The entire NETTRA-G2 program requires 146 K bytes of core storage, about 61 K being occupied by the actual program instructions and about 85 K by the stored data (compiled by FORTRAN H (OPT 2) compiler).

Fig. 3.1 shows the general organization of the program NETTRA-G2. It is identical to the organization of NETTRA-G1 (Fig. 2.1) except for the insertion of the new subroutine PROCIV. As in Fig. 2.1, an arrow from block i to block j indicates that the subroutine represented by block j is called by the subroutine represented by block i.

## 3.1 Flowchart of the Subroutines Realizing the Procedure

Although the procedure realized by NETTRA-G2 is quite different from that realized by NETTRA-G1, both of these programs use the subroutine

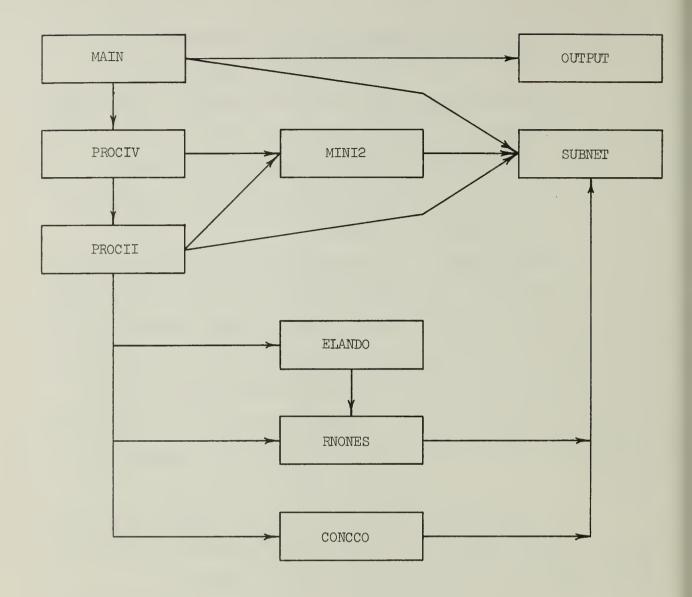


Fig. 3.1 General organization of the program NETTRA-G2.

PROCII to execute the main parts of the transformations. In NETTRA-G2, however, a new subroutine, PROCIV (for <u>PROCedure IV</u>, a name assigned during the development of the program), has been added to control the application of PROCII to the network.

PROCIV is a very simple subroutine. Its flowchart is shown in Fig. 3.1.1.

Block 1 of Fig. 3.1.1 calls MINI2 (a subroutine described in detail

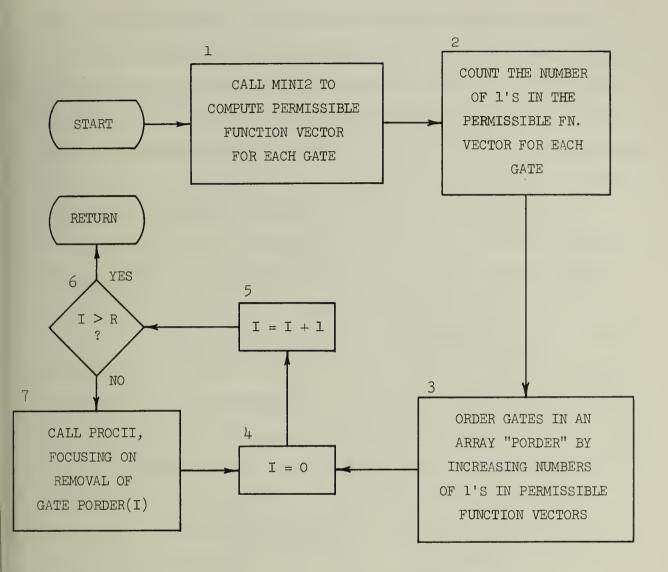


Fig. 3.1.1 Flowchart of PROCIV.

in [3]) to compute a permissible function vector for each gate. During this process some gates may be removed, but this probably will not happen unless the network contains a sufficient amount of redundancy. In any case, the purpose of calling MINI2 at this point is mainly to determine a set of permissible function vectors. Block 2 then counts the number of 1's appearing in the permissible function vector of each gate. This

information is used in block 3 to make an ordering of gates based on increasing numbers of 1's in the respective permissible function vectors of the gates. The ordered gate labels are stored in the array PORDER (such that PORDER(1) is a gate which has the least number of 1's in its permissible function vector, and PORDER(R) is a gate which has the greatest number).

Blocks 4, 5, 6, and 7 form a loop which repeatedly calls a special version of PROCII (the subroutine's characteristics are modified by specifying a certain parameter during the call to that subroutine). PORDER(1), PORDER(2), ... PORDER(R) are attempted to be removed from the network by the special PROCII - one gate, PORDER(I), upon each loop through block 7.

Each time PROCII is called specifying a particular gate, PORDER(I), the subroutine attempts the removal of that specific gate. The meaning of this will become clearer during the discussion of the flowchart of the "modified" PROCII. The reason for creating the ordering PORDER is to try to remove the more easily removable gates (gates with more 1's in their permissible function vectors are, in general, more essential to the network and thus are more difficult to remove) first, before they become more deeply entangled (e.g., by using their outputs as new connections to enable the removal of some other gate(s)) in the network.

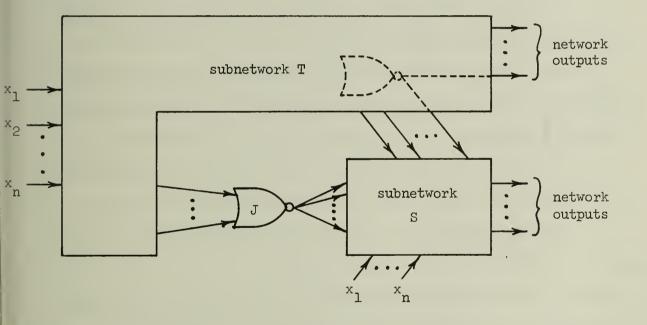
In the following discussion of the modified PROCII (let it be referred to as PROCII-M to distinguish it from the regular PROCII) a general knowledge of the information contained in [1] will once again be assumed.

Fig. 3.1.2 presents a diagram of a general multiple-output network [note the possibility that some of the network outputs of subnetwork T may be used as inputs to S (see gate drawn in dotted lines)] to be

transformed by PROCII-M in an attempt to remove the gate labeled "J".

(Arrows have been added only to indicate the direction of signal propagation through the connection lines.) Notice that all of the gates on every path from gate J to an output gate are contained in subnetwork S; every other gate in the network, besides J itself, is in subnetwork T. This distinction of whether each gate belongs to T or S is also made by the program itself when PROCII is modified to PROCII-M.

Fig. 3.1.2 Generalized network for the application of PROCII-M.



Generally PROCII-M operates in much the same way as PROCII, except:

(1) while gates in T (and external variables) may be connected to gates in S (or T) during the transformation, gates in S are not permitted to be

connected to any other gates; (2) covering of zeroes in gates of subnetwork T is not considered; (3) whenever there is a choice between the removal of a non-essential connection from a gate in T (or from an external variable) and one from a gate in S, the connection from the gate in S is removed; and (4) during the assignment of covering 1's (e.g., during block 23 of Fig. 2.1.2) external variables and gates in T are preferred covers. These modifications tend to push some of the burden of logic from subnetwork S to subnetwork T. In turn, this tends to reduce the necessity of gate J - hopefully until it is no longer needed at all. If this point can be reached, gate J may be removed from the network.

The above 4 differences are achieved by the following program modifications \* respectively:

- {1} In block 11 of the flowchart of PROCII (see Fig. 2.1.2) only external variables or the outputs of gates in T are allowed to become connectable functions.
- [2] In block 8 (see Fig. 2.1.1), if GCO is found to be a gate in subnetwork T, control of the program is sent back to block 3. If GCO is not in T, the regular tests (to see if GCO is an external variable or isolated gate) are then made.
- {3} Blocks 10 and 18 (Fig. 2.1.2) are the only parts of the program which eliminate non-essential inputs to each GCO. This is done in both blocks by calling the subroutine ELANDO. So difference (3) is effected by having

By specifying a certain parameter, covering of zeroes for gates in T can also be effected.

As previously mentioned, these modifications are built into the program and are trigered by the specification of the appropriate parameter during the call of PROCII. No reprogramming is required.

ELANDO first remove non-essential connections from gates in S to GCO. If any non-essential connections remain, they are then removed in the usual order (as in the normal PROCII).

[4] Block 23 (Fig. 2.1.2) makes the covering assignments. In choosing covers for zeros of GCO, block 23 prefers covers in the following general order: (a) external variables; (b) gates in subnetwork T; (c) gates in subnetwork S. Obviously, covers of type (a) or (b) place no requirements on the gates of subnetwork S.

This ordering, calculated in block 1 of Fig. 2.1.1 just after the determination of GORDER, is stored in the array RORDER. While the ordering of GORDER is still used in blocks 1 and 7 of the flowchart (Fig. 2.1.1), the ordering of RORDER replaces that of GORDER in making the covering assignments in block 23 (in Fig. 2.1.2). The ordering in this RORDER is divided into 3 groups of ordered gates (and external variables). The first group, stored in memory locations RORDER(1), ..., RORDER(k), is an ordering of all of the gates in T and all of the external variables (actually, the external variables are ignored in the use of this ordering although they do appear). The second group contains an ordering of all the gates in S.

This group can be found in locations RORDER(k+1), ..., RORDER(NR-1). Finally, the third group consists of only a single gate, J. It is stored in RORDER(NR). The overall preference of gates decreases from RORDER(1) through RORDER(NR).

After covers consisting of external variables have been assigned wherever possible, this preference ordering is employed to select covers for the remaining uncovered O components of GCO's permissible function vector.

Actually there are several different orderings for assigning covers which are available to the user. Only one of these will be described in detail here.

out the usual calculations (i.e., calculation of permissible function vectors, rearrangement of inputs, removal of redundant connections, etc.) for the gates of subnetwork T. This is accomplished by specifying a certain parameter while calling PROCII(-M). This parameter blocks the modification (2) discussed above. This might result in the removal of extra connections (and possibly, extra gates), but the necessary computation time is increased. The user should experiment to find whether or not this option is more suitable for solving his class of problems.

### 3.2. Examples for NETTRA-G2

As mentioned at the end of the previous section, with NETTRA-G2 the user has the option of allowing calculations (permissible function vectors, etc.) to be performed for all gates in the network for every PORDER(I) rather than just for gates in subnetwork S (with respect to the particular PORDER(I)). Examples both with and without this option will be given in this section. Beginning from the same initial network (Fig. 3.2.1), Fig. 3.2.2 and 3.2.3 show the results using NETTRA-G2, respectively, without and with the option.

The initial network, as documented by the printout displayed in Fig. 3.2.1 (a), utilizes 26 gates and 104 connections to produce a single 5-variable output function. Only uncomplemented variables are available as inputs to the network.

The output from NETTRA-G2 next shows the complete truth table for all of the gates of the original network (Fig. 3.2.1 (b)). This is followed by a description of the network's configuration (Fig. 3.2.1 (c)). Every

# Fig. 3.2.1 Initial Printout Describing a Network to be Transformed by NETTRA-G2.

## \*\*\*\*\*\* 5 VARIABLE, 1 OUTPUT TEST NETWORK

NUMBER OF VARIABLES = 5

NUMBER OF FUNCTIONS = 1

COST COEFFICIENT A = 100

B = 1

--- UNCOMPLEMENTED VARIABLES X ---

FUNCTION NO. 1. 1000010110001110010111

ORIGINAL NETWORK COST= 26104

(a) Heading information and network parameters.

#### TRUTH TABLE

```
1 = 1 0 0 0 0 1 0 1 1 0 0 0 1 1 1 0 1 1 0 0 0 0 0 1 1 1 0 0 1 0 1 1
12 = 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
13 = 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
14 = 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0
15 = 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0
16 = 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0
20 = 0 0 1 0 1 0 1 0 0 0 0 0 0 0 0 0 0 1 0 1 0 1 0 0 0 0 0 0 0
22 = 0 0 1 0 0 0 0 0 0 0 1 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
25 = 10001000100010001000100010001000100001000
```

(b) Truth table for original network.

```
GATE .. LEVEL
                  FED BY
        / 1/
                   3 4 5 6 8 9 13 15 17 19 20 22 24 26
  1
        / 3/
                  X1 X2 X3 X4 X5
  2
        / 2/
                  X1 X2 X3 X4 2
  3
       / 2/
  4
                  X1 X2 X3 2
        / 2/
  5
                  X1 X2 X4 X5
  6
        / 2/
                 X1 X2 X5 2
        / 3/
                 X1 X3 X4 X5
  7
  8
        / 2/
                 X1 X3 X4 7
       / 2/
  9
                 X1 X3 X5 2 7
        / 3/
 10
                  X1 X2
        / 3/
 11
                  X1 X4
        / 3/
 12
                 X1 X5
       / 2/
 13
                 X1 10 11 12
        / 3/
                 X2 X3 X4
 14
      / 2/
 15
                 X2 X3 14
      / 3/
                  X2 X3 X4 X5
 16
       / 2/
                  X2 X4 X5 16
 17
 18
        / 3/
                 X1 X2 X4
        / 2/
                 X2 X4 14 16 18
 19
        / 2/
                 X2 X5 2 16
 20
        / 3/
                 X3 X4 X5
 21
     / 2/
 22
                 X3 X5 21
        / 3/
                 X3 X4
 23
        / 2/
 24
                 X3 23
        / 3/
                 X4 X5
 25
        / 2/
 26
                 X4 11 14 18 23 25
```

<sup>(</sup>c) Configuration of original network.

# Fig. 3.2.2 Transformed Network Obtained by NETTRA-G2 Without Option (see text).

NETWORK DERIVED BY PROCIV
TIME ELAPSED= 217 CENTISECONDS

#### TRUTH TABLE

<sup>(</sup>a) Truth table for transformed network.

GATE	LEVEL	FED BY
1	/ 1/	8 13 20 24 26
2	/ 1/	
3	/ 1/	
14	/ 1/	
5	/ 1/	
6	/ 1/	
7	/ 3/	X5
8	/ 2/	X1 X3 7
9	/ 1/	
10	/ 3/	Х2
11	/ 3/	Xl
12	/ 1/	
13	/ 2/	X1 7 10 23
14	′ ′	
15	, ,	
16	, -,	X3 X4
17	/ 1/	
18	/ 1/	
19	, ,	
20	/ 2/	X2 X5 16
21	/ 1/	
22	/ 1/	
23	, -,	X <sup>1</sup> 4
24	/ 2/	хз 16
	/ 1/	
26	/ 2/	X4 7 11 16

<sup>\*</sup> A NETWORK DERIVED BY PROCIV COST= 11027

<sup>(</sup>b) Configuration of transformed network.

# Fig. 3.2.3 Transformed Network Obtained by NETTRA-G2 With Option (see text).

# NETWORK DERIVED BY PROCIV TIME ELAPSED = 320 CENTISECONDS

#### TRUTH TABLE

<sup>(</sup>a) Truth table for transformed network.

GATE	 LEVEL	FEI	) B	Z		
1	/ 1/	8	13	20	24	26
2	/ 1/					
3	/ 1/					
4	/ 1/					
5	/ 1/					
6	/ 1/					
7	/ 3/	X5				
8	/ 2/	Xl	ХЗ	7		
9	/ 1/					
10	/ 3/	X2				
11	/ 3/	Xl	Х7+			
12	/ 1/					
13	/ 2/	Xl	7	10	11	
14	/ 1/					
15	/ 1/					
16	/ 3/	Х3	Х7			
17	/ 1/					
18	/ 1/					
19	/ 1/					
20	/ 2/	X2	X5	16		
21	/ 1/					
22	/ 1/					
23	/ 1/					
24	/ 2/	Х3	16			
25	/ 1/					
26	/ 2/	X14	7	11	16	

<sup>\*</sup> A NETWORK DERIVED BY PROCIV COST= 10027.

<sup>(</sup>b) Configuration of Transformed Network.

gate is listed accompanied by a list of the gates and/or external variables which are its inputs.

If NETTRA-G2 is used without the option just discussed (i.e., when PROCIV calls PROCII for gate PORDER(I), calculations are carried out only for GCO ∈ S(PORDER(I))), the final result would be as shown in Fig. 3.2.2. The truth table for the transformed network appears in Fig. 3.2.2(a) (remember that outputs for disconnected gates are shown as all 1's). The required computation time for the transformation, 2.17 seconds, is also recorded. Fig. 3.2.2(b) contains a description of the corresponding network consisting of 11 gates and 27 connections.

However, if NETTRA-G2 is used with the option (i.e., when PROCIV calls PROCII for gate PORDER(I), calculations are carried out for every gate GCO in the network), the printed results would appear as in Fig. 3.2.3. Again, Fig. 3.2.3(a) shows the truth table corresponding to the transformed network, and Fig. 3.2.3(b) presents a description of the network's configuration. Note that in this case, the computation time has increased to 3.20 seconds while the network size has been further reduced to 10 gates and 27 connections.

#### 4. MAJOR FUNCTIONS OF COMMON SUBROUTINES

The subroutines realizing the 2 procedures presented in Sections 2 and 3 share the support of the following seven subroutines whose principal functions will be discussed in this section: MAIN, CONCCO, ELANDO, MINI2, OUTPUT, RNONES, SUBNET.

Complete program listings of these seven subroutines can be found in the appendix along with the listings of the subroutines realizing the previously described procedures.

The functions of the common subroutines are as follows:

MAIN: This subroutine repeatedly reads in groups of input data which include information about the given networks, e.g., the number of external variables, whether or not the complements of variables are also available as input variables, the number of output functions, the number of NOR gates, the list of connections, and the truth table of the output functions (see Section 5 for details). Using this information, MAIN constructs the incidence matrix, INC\$MX, for the network. INC\$MX is a two-dimensional array whose arguments represent gates or external variables. An array element INC\$MX(A1, A2) > 1 indicates a connection from Al to A2; an array element INC\$MX(A1, A2) < 0 indicates the absence of a connection from Al to A2. Next, subroutine SUBNET is called to calculate the level of each gate and to make lists of predecessors and successors (i.e., which gates precede which and which gates succeed which). MAIN then prints out the truth table and the constructed incidence matrix of the original network by calling the subroutine OUTPUT. Finally the desired transduction procedure is applied to the network by calling the subroutine(s) realizing the transformation. The configuration

of the transformed network is stored in INC\$MX, replacing the original network. Then MAIN prints the results of the transduction procedure, i.e., the elapsed time for the transformation, the new incidence matrix, and the new truth table.

CONCCO: During the execution of subroutine PROCII (which is involved in both of the transduction procedures described in this manual) sometimes a gate P (or a gate EFFCON) must be connected to a gate GCO (see blocks 15 and 21 of Fig. 2.1.2). Calling CONCCO (GCO, CCO) connects gate CCO to gate GCO and updates most of the arrays which are involved. Subroutine CONCCO has a second entry point, CNCCO. Calling CNCCO (GCO, CCO) has the same effect as calling CONCCO (GCO, CCO), except the call to CNCCO bypasses the update of the array INPTCV.

ELANDO: This subroutine, when called specifying the necessary argument GATE (e.g., CALL ELANDO(GATE)), eliminates non-essential (redundant) input connections to gate GATE and orders the remaining connections in a list L according to decreasing numbers of essential l's. This process, used in blocks 10 and 18 of the flowchart for PROCII, was discussed in detail in Section 2.1. If all of the input connections to a gate GATE are already known to be essential or if the non-essential connections are not to be removed, a call to ELANDO's entry point, ORDERL(GATE), will perform just the ordering function, making no attempt to search for and remove non-essential inputs to GATE (as in block 22 of PROCII, Fig. 2.1.2).

MINI2: MINI2 is a subroutine which realizes a pruning procedure (i.e., it transforms a network strictly by removing connections). In NETTRA-G1 and NETTRA-G2, it is used as a fast "clean-up" procedure to remove some of the redundant connections remaining after the main transformations have finished.

MINI2 is described in some detail in [3].

OUTPUT: This subroutine may be entered at five different points by a call to either OUTPUT, PAGE, LINE, TRUTH, or CKT.

OUTPUT assigns mnemonic names to external variables and gates for the purpose of achieving a readable print-out.

PAGE ejects one page on the printer.

LINE skips a specified number of lines on the print-out sheet. The number is specified by the argument in the call (e.g., "CALL LINE(5)" skips 5 lines).

TRUTH prints out the truth table of the network currently stored in INC\$MX.

CKT prints out the network configuration.

RNONES: Whenever an input TH is removed from a gate GCO, much work must be done to update the several arrays and variables involved. The bulk of this updating is performed by a call to this subroutine, CALL RNONES (GCO, TH, FLAG). The parameter FLAG (= 0, 1, or 2) is necessary since this subroutine is called from several different points in the program whose updating requirements differ slightly. The value of FLAG during the call to RNONES determines which particular variables and arrays will be updated.

SUBNET: This subroutine may be entered at three different points by a call to either SUBNET, UNNECE, or PVALUE.

SUBNET generates detailed information on the topology of the network stored in INC\$MX: (1) It calculates the level of each gate in the network. Level 1 is assigned to gates having no output connections (thus all gates which have been removed from the network will be assigned level 1). (2) It lists all immediate successors and immediate predecessors for each gate. (3) It calculates the successor matrix which is stored in a two-dimensional array, SUC\$MX. The value of SUC\$MX(A1, A2) indicates the existence or non-existence of a path from gate (or exteral variable) Al to gate A2.

UNNECE disconnects certain types of obviously unnecessary connections in the network and updates the above information (discussed in (1), (2), and (3)). The connections removed from the given network are those existing in no paths from the external variables to the output gates.

PVALUE calculates the actual truth table for the entire network stored in INC\$MX.

#### 5. INPUT DATA SETUP

In order to fully understand the description of the setup of the input data cards, certain preliminary explanations are necessary.

The purpose of network transductions is to reduce the cost of a network which realizes a certain function (or functions) or to alter the network in such a way as to allow another transduction to eventually accomplish such a reduction. This cost, C, is formally defined by the weighted sum of the number of gates, R, and the number of connections<sup>†</sup>, I, of a particular network, i.e.,

$$C = A \times R + B \times I$$

where weights A and B are arbitrary non-negative numbers.

Suppose the original network which is to be transformed produces m output functions of n variables. Let  $\mathbf{x}_{\ell}$ ,  $\ell=1,\ldots,$  n, be the external variables and  $\mathbf{f}_{\mathbf{h}}$ ,  $\mathbf{h}=1,\ldots,$  m, be the output functions. Before a transformation can be performed on a network by a program, a description of that network must be input to the program. In the case when all of the output functions are completely specified (i.e., no "don't cares"), specifying only the interconnection pattern of the network is sufficient. But if one or more of the output functions is not completely specified, then the user must also provide the truth table (truth tables for all output functions are condensed into a single table) of the problem. Providing the truth table to the program consists of two steps,

t A "connection" refers to either a connection from an external variable or an interconnection between two gates.

namely the specification of external variables, and the specification of output functions.

The method of specifying the output functions depends directly upon the method chosen to specify the external variables. External variables may be specified in either of two ways, (a) an implicit specification of external variables, or (b) an explicit specification of external variables.

(a) In the case of implicit specification of external variables, the user specifies the number n of external variables along with a parameter which indicates whether or not the uncomplemented variables are available. Reading the value n along with the parameter, the program internally generates the entries of external variables of an ordinary truth table, that is, a truth table which consists of 2<sup>n</sup> input vectors, as shown in Fig. 5.1. In this truth table, the input vectors are arranged according to the order such that an integer j expressed in a binary representation (x<sub>1</sub>...x<sub>n</sub>) increases, where x<sub>1</sub> is the most significant digit and x<sub>n</sub> is the least significant digit. For example, the truth table for a function of three variables is shown in Fig. 5.2.

The implicit specification of external variables is used for logical design problems in which the output functions have relatively few don't-care terms.

	( x <sub>1</sub>	$x_1^0 \dots x_1^j \dots x_1^{2^n-1}$
The uncomplemented variables		•
	x <sub>n</sub>	$x_n^0 \cdots x_n^j \cdots x_n^{2^n-1}$
	$\left( \frac{\overline{x}}{x_1} \right)$	$\vec{x}_1^0 \dots \vec{x}_1^j \dots \vec{x}_n^{j-1}$
The complemented variables	<b>\ :</b>	•
	$\left\lfloor \frac{\overline{x}}{n} \right\rfloor$	$\bar{x}_n^0 \dots \bar{x}_n^j \dots \bar{x}_1^{2^{n-1}}$
	fl	$f_1^0 \dots f_1^j \dots f_1^{2^n-1}$
The output functions	f <sub>2</sub>	
	•	•
	f <sub>m</sub>	$f_m^0 \dots f_m^j \dots f_m^{2^n-1}$

These entries exist only in the case of logical design problems where the complemented variables are available as external inputs.

Fig. 5.1 The truth table of output functions of n variables

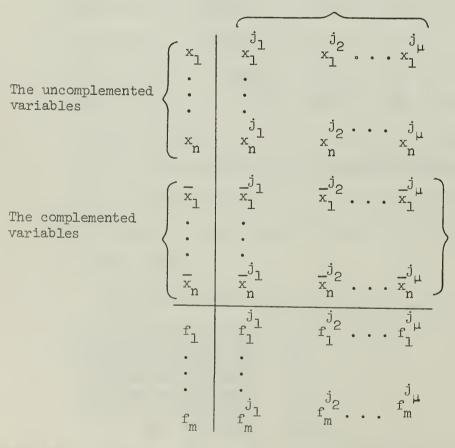
f	f.	)					•	f <sub>1</sub>	
<del>x</del> 3	1	0	1	0	1	0	1	ر ٥	
$\bar{x}_2$	1	1	0	0	1	1	0	0	
$\bar{x}_1$	1	1	1	1	0	0	0	0	-
*3	0	1	0	1	0	1	0	1	
x <sub>2</sub>	0	0.	1	1	0	0	1	1	
×1	0	0	0	0	1	1	1	1	

These entries exist only in the case of logical design problems where complemented variables are available as input variables.

ig. 5.2 The truth table of a function of three variables.

(b) In the case of explicit specification of external variables, the user specifies the entries of external variables of the truth table using additional cards called < external-variable-card > s. The explicit specification of external variables is used in the case of logical design problems where output functions have many don't-care terms. Suppose that the output functions are defined for a subset of input vectors of the entire truth table of Fig. 5.1. Let  $\vec{x}$   $\vec{j}$ ,  $j = j_1$ ,  $j_2$ , ...,  $j_{\mu}$  denote these input vectors. The user can 'condense' the truth table of Fig. 5.1 into another table shown in Fig. 5.3.

only µ input vectors



These entries exist only in the case of logical design problems where the complemented variables are avaiable as external inputs.

Fig. 5.3 A 'condensed' truth table having only the input vectors  $\vec{x}^j$ ,  $j = j_1$ , ...,  $j_{\mu}$ , for which the output functions are defined.

Using < external-variable-card > s, the user can set up internally the table of Fig. 5.3 in place of Fig. 5.1.

### 5.1 Input Data Card Format

For each separate problem, a set of input data cards must be submitted which consists of the following †:

- (i) < heading-card >
- (ii) < problem-parameter-card >
- (v) < connection-description-card > s

Both (i) and (ii) will always consist of only a single card, but (iii), (iv), and (v) may each consist of several cards. Furthermore, types (iii) and (iv) are omitted if all output functions are completely specified, and (iii) need only be prepared in the case of the explicit specification of external variables for the truth table. Following is a description of the formats for each type of input card, (i), (ii), (iii), (iv) and (v):

## (i) < Heading-card >

This is the first card of the input deck for a problem. This card may contain any alphanumeric information, in columns 1~80, which may be used for the identification of the problem, but none of the information on this card will be used in the actual computation. This information will be printed on the first page of the output.

t The current implementations of the NETTRA programs accept only heading, problem-parameter, and connection-description cards. Eventually it is hoped that these programs will be modified to accept all of the options described in this section.

### (ii) < Problem-parameter-card >

This card specifies the nature of the problem the user wants to solve. There are 7 fields in which to specify the parameters with characters and numerals. These fields are as follows:

Cols. 1~4: An integer, N, which is right-justified.

This number, N, represents the number of external variables, n, of the output functions. Be sure to punch n (rather than 2n) for N in the case of both complemented and uncomplemented variables available.

Cols. 5~8: An integer, M, which is right-justified.

This number, M, is the number of output functions, m, to be realized simultaneously. Therefore, of course, M will also be the number of output gates in the network.

Cols. 9~12: An integer, R, which is right-justified.

This number, R, specifies the number of gates which are included in the network. For various reasons, the user may wish to input networks in which one or more of the gates are "isolated" (i.e., are not connected to any other gates). This is permissible as long as these "isolated" gates are also included in the total number of gates, R.

Cols. 13~16: An integer, A, which is right-justified.

The number A is the value of the non-negative weight for the number of gates in the cost function. (See Table 5.1.1, 'Typical combinations of values A and B for different network reduction problems'.)

Cols. 17~20: An integer, B, which is right-justified.

The number B is the value of the non-negative weight for the number of connections in the cost function. (See Table 5.1.1.)

Col. 24: A blank 'b', or one of the characters, 'C', 'X', 'Y', 'U' or 'V'.

The 'b' or 'C' parameter represents an implicit specification

of both the external variables and an implicit specification of

the output functions (in this case, the output functions will be

calculated from the connection pattern of the network). The 'X' or

'Y' parameter indicates an implicit specification of external variables

only. The 'U' or 'V' parameter indicates an explicit specification of

external variables. (See summary of these symbols in Table 5.1.2)

The 'b' or 'X' parameter specifies that only uncomplemented external variables are available for the network. The 'C' or 'Y' parameter specifies that both uncomplemented and complemented variables are available for the network. If the user specifies the 'b', 'X', 'C', or 'Y' parameter, the program sets up the truth table by generating a set of  $2^n$  input vectors  $(x_1^j, \ldots, x_n^j)$ , for  $j=0, \ldots, 2^n-1$ , in the case of a 'b' or 'X' parameter, or  $(x_1^j, \ldots, x_n^j, \overline{x_1^j}, \ldots, \overline{x_n^j})$  for  $j=0, \ldots, 2^n-1$ , in the case of a 'C' or 'Y' parameter.

The 'b' or 'C' parameters should be used for problems in which the output functions contain no don't-care terms. For such problems, the preparation of the < external-variable-card > s and the < output-function-card > s can be dispensed with since the program can calculate completely all output functions using only a description of the

<sup>†</sup> A 'b' stands for a blank (i.e., no character punched).

Network Reduction Problem	Values of A and B
reducing the number of gates only.	A = 1 and $B = 0$
reducing the number of gates primarily, then reducing the number of connections secondarily.	A = 100 and B = 1
reducing the number of connections only.	A = 0 and $B = 1$
reducing the number of connections primarily, then reducing the number of gates secondarily.	A = 1  and  B = 100
reducing the sum of the number of gates and the number of connections.	A = B = 1

Table 5.1.1 Typical combinations of values A and B for different network reduction problems.

t Most of the programs in the NETTRA system are oriented toward this reduction problem, so the user will probably find this combination of A and B the most useful.

uncomplemented variables only available	both complemented and uncomplemented variables available	
'b'	'C'	implicit specification of external variables and output functions
'X'	'Y'	implicit specification of external variables
'U'	'ν'	explicit specification of external variables

Table 5.1.2 Possible symbols for column 24 of < problem-parameter-card >.

connection pattern of the network (provided by the <connection-description-card>s).

Similarly, the 'X' or 'Y' parameter implies the use of a complete truth table (i.e., 2<sup>n</sup> input vectors for n external variables) inside the program. Since from this information the program can easily generate the truth table entries for the external variables, as just mentioned, the < external-variable-card > s are unnecessary.

The m < output-function-card > s, however, must still be prepared.

The 'U' parameter specifies that only uncomplemented external variables are available for the network. The 'V' parameter specifies that both uncomplemented and complemented variables are available for the network. In either case, the 'U' or the 'V' parameter, the user <u>must</u> prepare n < external-variable-card > s and m < output-function-card > s. The program sets up the truth table by reading these < external-variable-card > s and < output-function-card > s.

Cols. 25~28: An integer, NEPMAX, which is right-justified.

This parameter is omitted for all NETTRA programs except those involving "error-compensation" routines. In the cases where NEPMAX is required, a further discussion of this parameter can be found elsewhere in the manual. The abbreviation NEPMAX is a mnemonic for "maximum number of error positions", and the default is  $NEPMAX = 2^{(n-1)}, \text{ where n is the number of external variables.}$ 

## (iii) < External-variable-card > s

In combination with the 'U' or 'V' parameter in column 24 of
the < problem-parameter-card >, the n < external-variable-card > s
specify the entries of external variables of the truth table of

Fig. 5.3. Each card contains the binary representation of external variable  $\mathbf{x}_{\ell}$ , i.e.,  $(\mathbf{x}_{\ell}^{\mathbf{j}1}, \mathbf{x}_{\ell}^{\mathbf{j}2}, \ldots, \mathbf{x}_{\ell}^{\mathbf{j}\mu})$ , starting from column 1 of the card. The maximum number of bits in a binary representation is limited to 32. (This means the maximum number of input vectors is 32.) If the actual number of bits is less than 32, then a termination symbol '/' (slash) is put on the right of the right-most bit of the binary representation on the first < external-variable-card >. The remaining columns after the termination symbol '/' in the first card, as well as the same columns in the following cards, may contain any alphanumeric information which may be used for identification. This information will not be printed on the output pages.

In the case of the 'V' parameter, the program generates the binary representations corresponding to complemented variables by taking negations of the entries of the < external-variable-card > s. Therefore the user <u>must not</u> provide < external-variable-card > s representing the complemented variables,  $\bar{x}_{\ell}$ .

If one of the parameters 'b', 'C', 'X', or 'Y' appears in column 24 of the < problem-parameter-card >, the user does not prepare < external- variable-card > s.

## (iv) < Output-function-card > s

The m < output-function-card > s specify the set of m output functions to be realized simultaneously. Each card contains the binary representation of one output function  $f_h$ , starting from column 1 of the card. A symbol '\*' is used to denote don't-care terms, if any. The maximum number of bits in a binary representation is limited to 32.

The actual number of bits must be 2<sup>n</sup> in the case of an implicit specification of external variables, or must be the same as defined by the < external-variable-card > s in the case of an explicit specification of external variables. The remaining columns, up to column 72 (inclusive), may contain any alphanumeric information which may be used for identification. This information will not be printed on the output pages.

If either the 'b' or 'C' parameter appears in column 24 of the < problem-parameter-card >, the < output-function-card > s must be omitted.

### (v). < Connection-description-card > s

In the present version of the program, 9 cards (some of which may be just blank cards) are required. Each of these cards is divided into 16 fields of 5 columns each (i.e., columns  $1 \sim 5$ ,  $6 \sim 10$ ,  $11 \sim 15$ , ...,  $71 \sim 75$ ,  $76 \sim 80$ ). Beginning with the first field of the first card, continuing through the succeeding fields of that card and through the fields of as many additional cards as necessary (up to a maximum of 9, total), the expressions (explained in the next paragraph)  $C_1$ ,  $C_2$ ,  $C_3$ , ..., are punched right-justified in their respective fields.

Each gate of the network is labeled uniquely by assigning it one of the integers 1, 2, ..., R, such that the output gates receive

t For many uses, the user will probably find that these 9 cards far exceed his needs, and may thus be inconvenient. In such a case, the number of required cards may be easily adjusted by making the obvious changes in two statements (A READ statement and a DO statement) following the comment card "C\*\*\*\* READ IN NETWORK INFORMATION AND SET UP INC\$MX \*\*\*\*\*" in subroutine MAIN.

the labels 1, 2, ..., m. The names X1, X2, ..., Xn are assigned to the external variables  $\mathbf{x_1}$ ,  $\mathbf{x_2}$ , ...,  $\mathbf{x_n}$  (and the names Y1, Y2, ..., Yn to the complemented external variables  $\overline{\mathbf{x_1}}$ ,  $\overline{\mathbf{x_2}}$ , ...,  $\overline{\mathbf{x_n}}$ , if appropriate). \(^{\frac{1}{2}} Now, for each connection of the network (i.e., including both the connections from external variables to gates and connections from gates to other gates), a \(^{\frac{1}{2}} character expression,  $\mathbf{C_i}$ , is formed, to represent that connection as follows: to represent a connection from gate GI to gate GJ, the numeric label GI is inserted into the first two character positions of  $\mathbf{C_i}$  and the numeric label GJ is inserted into the second two positions (e.g., the  $\mathbf{C_i}$  for a connection from gate 9 to gate 5 would be "0905"); to represent a connection from external variable XI to gate GJ, the alphanumeric label XI is inserted into the first two character positions of  $\mathbf{C_i}$  and the numeric label GJ into the second two positions (e.g., the  $\mathbf{C_i}$  for a connection from external variable  $\mathbf{x_2}$  to gate 10 would be "X310").

Every connection of the network must be represented by a  $C_i$ , although there are no restrictions on the order in which the connections (i.e.,  $C_i$ 's) are punched onto the input cards.

that the time of writing, the programs have not yet been changed to recognize this new labeling system. The old labels are simply:

1, 2, ..., n, for external variables  $x_1$ ,  $x_2$ , ...,  $x_n$  (and n+1, n+2, ..., 2n for the complemented variables  $\overline{x_1}$ ,  $\overline{x_2}$ , ...,  $\overline{x_n}$ , if they are permitted in the problem); n+1, n+2, ..., n+m for the m output gates of the network  $(2n+1, 2n+2, \ldots, 2n+m)$  if complemented variables are included); and finally n+m+1, n+m+2, ..., n+R (2n+m+1, 2n+m+2, ..., 2n+R) for the non-output gates of the network.

These five groups of cards, (i), (ii), (iii), (iv) and (v) in the correct order constitute the necessary description for a single problem. In order to solve several problems during the same computer run, the descriptions for the desired problems are just arranged serially. See Fig. 5.1.1 for an example of the sequencing of all cards for the execution of a NETTRA program using typical JCL statements for the IBM 360/75.

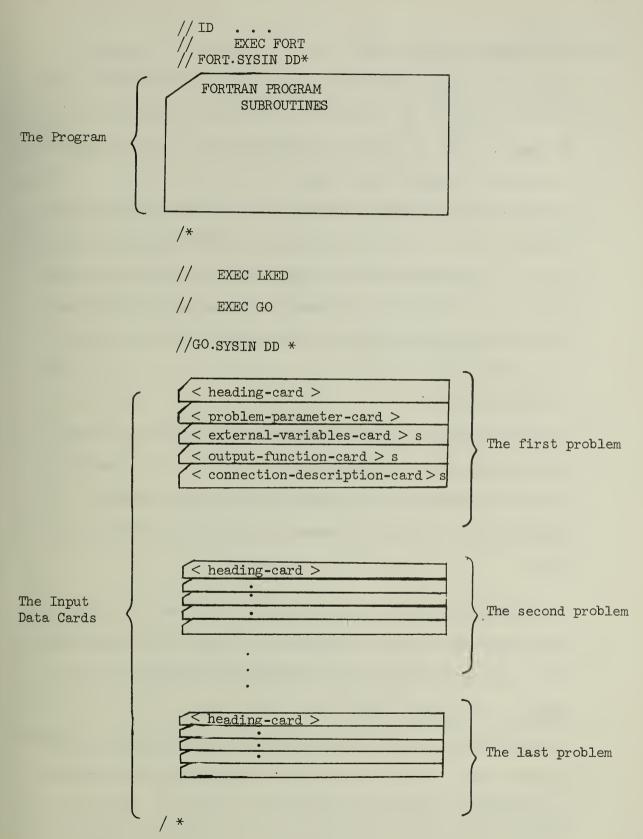


Fig. 5.1.1 Input card sequence for the execution of a typical NETTRA program.

### 5.2 Restrictions on Problem Size

The number n of external variables.

< external-variable-card > s.

The number R of gates.

2.

3.

40-2n.

In order to fit the programs into a finite amount of space, some restrictions on the size of an acceptable problem are required:

- 1. The number t of input vectors in the truth table is 32 or less.
- Because of  $t \le 32$ , n is 5 or less in the case of completely specified functions. In the case of incompletely specified functions, however, any  $n \le 20$  is acceptable if only uncomplemented variables are available, or  $n \le 10$  if both uncomplemented and complemented variables are available, provided that the truth table is defined by the
- The number of gates, R, may not exceed 40-n in the case of only uncomplemented variables available (a 'b', 'X', or 'U' parameter).

  In the case of both uncomplemented and complemented variables available (a 'C', 'Y' or 'V' parameter), the limit is lowered to

All of these limitations are essentially imposed by the array sizes in the programs as presently written. To loosen the restrictions is mainly a task of increasing array dimensions appropriately.

## 5.3 Examples of Input Data Setup

The following examples will illustrate, for the general program in the NETTRA system, various possible input data card setups complying with the directions given in Section 5.1.

Example 1: A two output network of four variables shown in Fig. 5.3.1. Assume the two output functions are  $f_1 = CCEF^{\dagger}$  and  $f_2 = 3BBB$  and only uncomplemented variables are available. Furthermore, assume it is desired to reduce the number of gates primarily and the number of connections secondarily (see Table 5.1.1).

From this description, the < problem-parameter-card > must contain the following values:

Cols. 1~4 4, the number of external variables Cols. 5~8 2. the number of output functions 8, Cols. 9~12 the number of gates in the original network Cols. 13~16 100, the value of A Cols. 17~20 1, the value of B 24 'b', uncomplemented variables only available and Cols. implicit specification of both the external variables and the output functions Cols. 25~28 'b', since the NEPMAX parameter is unrelated to the program to be used

Fig. 5.3.2 shows the setup of data cards used to specify the network in Fig. 5.3.1 as input for the program. Notice that in forming the C<sub>i</sub>, the four uncomplemented variables are represented by the labels X1, X2, X3, X4; the two output gates by the numbers 1, 2; and the remaining gates, by the numbers 3, 4, 5, 6, 7, 8. This manner of labeling is

t For convenience, the output functions are expressed in hexidecimal notation. When the numbers in this notation are expanded into binary, they represent the output vectors as they appear (i.e., in the same left-to-right order) in the complete truth table described earlier and pictured in Fig. 5.1.

<sup>††</sup> This assumption is implicit in most of the transduction procedures and their implementations which comprise the NETTRA system.

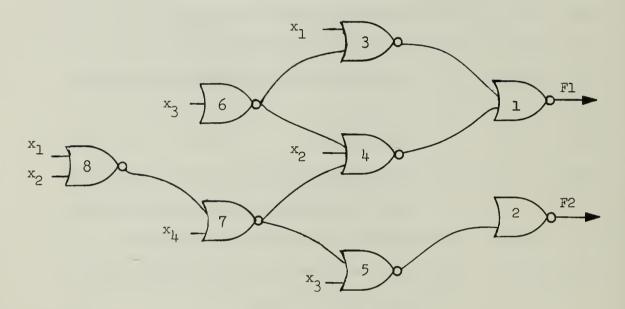
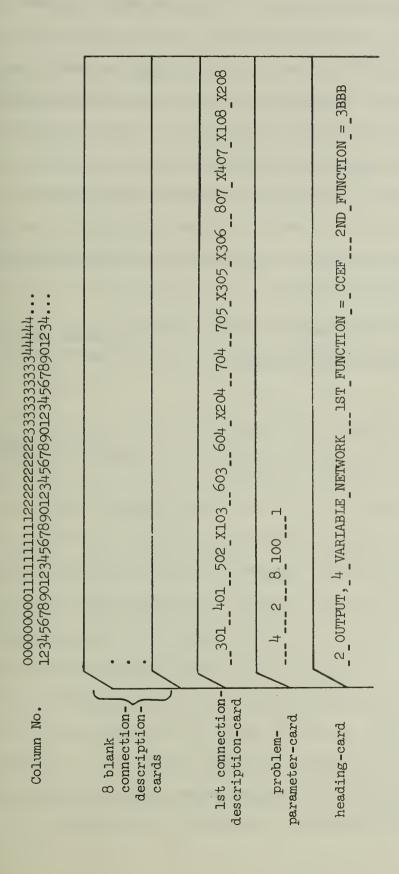


Fig. 5.3.1 Network to be transformed in Examples 1 and 2.



Possible setup of data cards to specify the problem given in Example 1. Fig. 5.3.2

strictly required by the instructions for preparing the < connectiondescription-card > s (see Section 5.1).

The heading card in Fig. 5.3.2 will simply be read by the program and printed character for character onto the output page as an identification of the particular problem. Below that, the number of variables, number of functions, and the cost coefficients, A and B, will be printed (all with appropriate labels). Also, immediately following will be a statement of what types of external variables are permitted (i.e., either just uncomplemented variables or both complemented and uncomplemented) along with their generic names:

X - for uncomplemented variables

Y - for complemented variables

if external variables were implicitly specified

or

U - for uncomplemented variables

·V - for complemented variables

if external variables were explicitly specified

For example, if both X and Y appear as generic names (as would occur in the case of an implicit specification of external variables with both complemented and uncomplemented variables available) then the external variable names which appear on subsequent output pages will be X1, X2, ..., Xn and Y1, Y2, ..., Yn. Or, if both U and V appear as generic names (as would occur in the case of an explicit specification of external variables with both complemented and uncomplemented variables available) the external variable names which appear in the output will be U1, U2, ..., Un (for the uncomplemented variables) and V1, V2, ..., Vn (for the complemented variables). It should be noted, however, that the letters U and V, as used as replacements for X and Y (respectively) in the

naming of external variables (e.g. Ul, Vl instead of Xl, Yl), appear strictly on the output pages of the program - they are <u>not</u> used internally in the program and they <u>must not</u> appear in the variable names punched on the < connection-description-card > s by the user. They are intended only as an aid to the user so that, at a glance at the transformed network in the output, he can easily distinguish whether the external variables were implicitly or explicitly specified for that particular problem.

Following the statement of whether only uncomplemented or both complemented and uncomplemented external variables are employed, the user will find next on the output page the cost of the original network which was input to the program. This is the cost which was defined in the beginning of Section 5.

The cost will be followed by a truth table (generally in the same form as Fig. 5.1) showing the outputs (0 or 1) of all of the gates in the network for every external variable input combination (i.e., combinations of 0's and 1's) of interest.

Finally, below the truth table will be printed a description of the network submitted as input. This is for documentation purposes, and it is also much more readable than the network description which appeared on the < connection-description-card > s. In this description, each gate is listed along with the names of the gates and external variables which feed it. Also, to assist the user in sketching the network from its description, the level of each gate in the network is included (gates which do not feed other gates are assigned to level 1, all other gates are assigned level numbers such that each gate is in a level one

higher than the highest level gate directly fed by it).

All of the information just described will be printed before the execution of the transduction actually begins. This will be followed, beginning at the top of a new output page, by the network(s) actually obtained as a result of the computation. First the complete truth table of the transformed network will be printed, followed by a network connection description of the form just described above. Finally, the cost of the new network will be calculated and printed.

In this example, it was assumed that there were no "don't-cares" in the output functions implicitly specified by the input, thus no < external-variable-card > s or < output-function-card > s were included.
In the next example, however, < output-function-card > s will be required in order to specify some of the components of the output functions as "don't-cares".

Example 2: The two output network of four variables shown in Fig. 5.3.1. This is the same network used in Example 1, but this time the output functions are not assumed to be completely specified. Let  $f_1 = \text{'ll00l**0l*l0*lll'}$  and  $f_2 = \text{'0**ll0lll*lll0ll'}$  be the required functions. Also, suppose that both complemented and uncomplemented variables are desired to be available during the transduction. Again the problem is to reduce the number of gates primarily and the number of connections secondarily.

For this problem, the following values must appear on the < problem-parameter-card >:

Cols. 1~4 4, the number of external variables

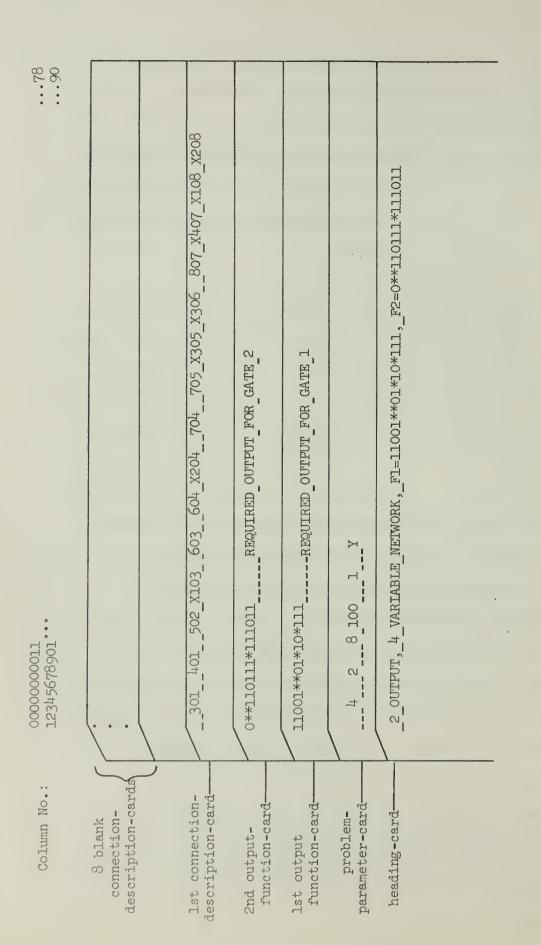
Cols. 5~8 2, the number of output functions

Cols. 9~12 8, the number of gates in the original network
Cols. 13~16 100, the value of A
Cols. 17~20 1, the value of B
Col. 24 Y, indicative of an implicit specification
of external variables and the availability
of both complemented and uncomplemented
variables

Fig. 5.3.3 shows the setup of the data cards corresponding to this problem. Notice the differences and similarities to the data cards shown in Fig. 5.3.2. The < problem-parameter-card > differs only in column 24. The < external-variable-card > s are missing in both Fig. 5.3.2 and Fig. 5.3.3 since the external variables are implicitly specified for both problems. The < output- function-card > s, however, appear in Fig. 5.3.3 but not in 5.3.2 since they are necessary to specify "don't-care" components which do not occur in the completely specified output functions of Example 1. In both cases, though, the < connection-description-card > s are identical since the original networks are identical.

By allowing "don't-care" terms in the output functions, and by allowing the use of both complemented and uncomplemented variables (even though the original network employed only uncomplemented variables), the restrictions during the transduction process are loosened (compared to what they were for Example 1), perhaps permitting a network of

t In the case of NETTRA-PG1, -P1, and -P2, it is useless to specify Y rather than X in column 24 for this example. Since the original network uses only uncomplemented variables, to these programs which perform "pruning" procedures (i.e., procedures which are incapable of adding new connections) the availability of complemented variable is not meaningful.



Possible setup of data cards to specify the problem given in Example 2. Fig. 5.3.3

less cost to be obtained.

Notice that the first < output-function-card > corresponds to the output of gate 1 and the second < output-function-card > corresponds to the output of gate 2. This must hold true for every problem in which < output-function-card > s are included; the gates labeled 1, 2, ..., m must correspond to the output functions specified on < output-function-card > s 1, 2, ..., m, respectively.

Of course, the printed output of the program will be in the same format described in Example 1.

Example 3: The three output network of six variables shown in Fig. 5.3.4. The outputs are again assumed to be incompletely specified. In fact, only the following 11 input combinations are specified out of a possible  $64 \ (= 2^6)$ :

×ı	0	0	0	0	0	0	0	0	0	0	1
x <sub>2</sub>	0	0	0	0	0	0	0	1	1	1	0
ж <sub>3</sub>	0	0	0	0	0	0	0	0	0	1	1
×4	0	0	0	0	1	1	1	0	0	0	1
× <sub>5</sub>	0	0	1	1	0	0	1	0	1	1	0
×6	0	1	0	1	0	1	0	1	1	0	0
Fl	0	0	1	1	0	0	*	0.	0	0	0
F <sub>2</sub>	1	1	*	1	1	1	0	1	1	0	*
F <sub>3</sub>	1	ļ	0	0	0	0	0	1	0	0	0

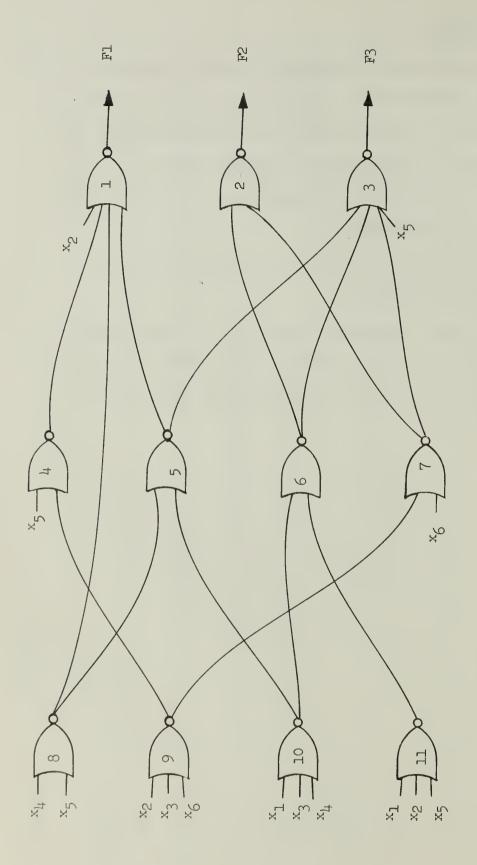


Fig. 5.3.4 Network to be transformed in Example 3.

Additionally, only uncomplemented variables are assumed to be available, and the problem is to reduce the number of gates primarily and the number of connections secondarily.

For this example, the following parameters appear on the < problem-parameter-card >:

Cols. 1~4 6, the number of external variables

Cols.  $5 \sim 8$  3, the number of output functions

Cols. 9~12 11, the number of gates in the original network

Cols. 13~16 100, the value of A

Cols. 17~20 1, the value of B

Col. 24 U, indicative of an explicit specification of external variables and the availability of only uncomplemented variables

Fig. 5.3.5 shows a possible setup of the data cards corresponding to this example. Notice that in this example, the <external-variable-card > s are included, whereas in the two previous examples they were omitted. Although this problem is not too realistic (none of the 3 functions is actually a 6-variable function), it demonstrates the input data preparation to be used in cases where many external variables are present and a high percentage of "don't care" terms exist.

Again, the printed output from the program will follow the same format described in Example 1.

Column No.: 1234567890123\*\*\*

\_x201\_\_401\_\_501\_\_801\_\_602\_\_702\_x503\_\_503\_\_603\_\_703\_x504\_\_904\_\_805\_1005\_1006\_1106 LIGX\_LIGX\_LILX\_014X\_015X\_01LX\_609X\_609\_X309\_X508\_X408\_X408\_709X\_ 2 댔 X FUNCTION F3 FUNCTION FI VARIABLE\_X2 X<sup>t</sup> VARIABLE\_X1 \_3\_OUTPUT, 6\_VARIABLE\_NETWORK VARIABLE VARIABLE VARIABLE VARIABLE FUNCTION 001\_11\_-0000000000 0001700\*0000 /100000000000 11000001000 0011001100 0101010100 110000000000 \*011011\*11 00001110000 9 connection-descripoutput-functionproblem-parameterconnectiondescriptionexternalvariableheading-card first two 7 blank tion-cards cards cards

Possible setup of data cards to specify the problem given in Example 3. Fig. 5.3.5

## REFERENCES

- [1] Y. Kambayashi and J.N. Culliney, "NOR network transduction procedures based on connectable and disconnectable conditions (Principles of NOR network transduction programs NETTRA-Gl and NETTRA-G2)," to appear as a Report, Dept. of Comp. Sci., Univ. of Ill., Urbana, Ill.
- [2] Y. Kambayashi and S. Muroga, "Network transduction based on permissible functions (General principles of NOR network transduction NETTRA programs)," to appear as a Report, Dept. of Comp. Sci., Univ. of Ill., Urbana, Ill.
- [3] H.C. Lai and J.N. Culliney, "Program manual: NOR network pruning procedures using permissible functions (Reference manual of NOR network transduction programs NETTRA-PG1, NETTRA-P1, and NETTRA-P2)," Report No. UIUCDCS-R-74-686, Dept. of Comp. Sci., Univ. of Ill., Urbana, Ill., Nov. 1974.

## APPENDIX:

## Program Listings

Following are the listings of the FORTRAN programs NETTRA-Gl and NETTRA-G2. These programs realize, respectively, the transduction procedures discussed in Sections 2 and 3.

Since NETTRA-G2 differs just in two subroutines from NETTRA-G1, only those two subroutines are listed for NETTRA-G2: MAIN (differs only slightly from MAIN in NETTRA-G1) and PROCIV (not contained in NETTRA-G1).

Explanations of variables used in the programs can be found in the listings themselves.

```
GGG
                                                      RRRR
            PPPP
                       RRRR
                                  000
C.
                 P
                       R
                            R
                                 0
                                      0
                                                      R
                                                                                  MM
                       R
                            R
                                 0
                                      0
                                            G
                                                      R
                                                           R
            PPPP
                       RRRR
                                 n
                                            G
                                               GG
                                                      RRRR
                                                                 4444
                                      0
                                            G
                                                      R
                                                                                   M
                                 0
                                      0
                                                 G
                                                                           M
                                  000
                                             GGG
C
                                                                            GGG
            FEFFE
                       TTTTT
                                 TTTTT
                                            RERE
                                                                                       1
  N
                                                 R
C
                                    T
                                            R
  NN
                         T
                                                                           G
                                                                                G
                                                                                      11
       N
            E
C
                                            R
                                                 R
                                                                           G
            Е
                         T
                                    T
  NNN
C
            EEE
                         T
                                    T
                                            RRRR
                                                      AAAAA
                                                                 XXXXX
                                                                           G
                                                                               GG
                                                                                       1
  N
      NN
C
                         T
                                    Т
                                            R
  N
       N
            E
                                                                                G
                                                                                       1
C
                                                                            GGG
       N
            EEEEE
                                                                                      111
5
       IMPLICIT INTEGER * 4 (A-T, V-Z, $), REAL (U)
C
       NOTE: ALL COMMON VARIBLES MIGHT NOT BE USED IN THIS PROGRAM.
```

0000

C

CCC

C

C

```
G1 00010
G1 00030
                                                                   G1 00040
COMMON VARIABLES:
                                                                   G1
                                                                      00050
                                                                   G1 00060
   $GT: POINTS TO A 'COLUMN' OF POTAB. FOR EACH 'ROW' THE ENTRY
  IN THIS COL. TELLS GATE WHERE FN. IS REALIZED.
$LTH: POINTS TO A 'COLUMN' OF POTAB. FOR EACH 'ROW' THE ENTRY
                                                                   G1 00070
                                                                   G1
                                                                      00080
        IN THIS COL. TELLS HOW MANY CONNECTIONS MUST BE ADDED.
                                                                   G1
                                                                      00090
  SNOE: POINTS TO A "COLUMN" OF POTAB.
                                        FOR EACH 'ROW' THE ENTRY
                                                                   G1 00100
        IN THIS COL. TELLS THE NUMBER OF 1-ERRORS CREATED IF THIS
                                                                   G1
                                                                      00110
        ROW IS USED.
                                                                   G1 00120
       POINTS TO A 'COLUMN' OF POTAB. FOR EACH 'ROW' THE ENTRY
                                                                   G1 00130
        IN THIS COLUMN TELLS THE PREFERENCE WEIGHT.
                                                                   G1 00140
     A: WEIGHT FOR NO. OF GATES IN COMPUTING COST FUNCTION.
                                                                   G1 00150
     B: WEIGHT FOR NO. OF CONNECTIONS IN COMPUTING COST FUNCTION.
                                                                   G1 00160
  COST: COST OF NETWORK - A MEASURE OF NETWORK SIZE.
                                                                   G1 00170
 ESSIS: RECORDS NO. OF ESSENTIAL 1°S IN EVERY INPUT TO CURPENT GCOG1 00180
        (POSITIONS IN ESSIS CORRES. TO GATES NOT FEEDING GCO ARE
                                                                   G1 00190
        IGNORED).
                                                                   G1 00200
 FSUB1: POINTS TO LAST ELEMENT IN F$1.
                                                                   G1 00210
   F$1: LISTS (CONSECUTIVELY) POSITIONS OF DESIRABLE 1'S (FOR
                                                                   G1 00220
        COVERING) IN A CONNECTIBLE FUNCTION.
                                                                   G1 00230
       LABEL OF A PARTICULAR GATE.
                                                                   G1 00240
GLEVEL: GLEVEL(GI) TELLS WHICH LEVEL OF THE NETWORK GI IS IN.
                                                                   G1 00250
GSMALL: STORES INTERMEDIATE AND FINAL CALCULATED CSPF'S.
                                                                   G1 00260
 HLIST: HLIST(I, J) GIVES NAME OF I-TH GATE (QR EX. VAR.) IN NET-
                                                                   G1 00270
        WORK LEVEL J.
                                                                   G1 00280
  IDXO: LIST OF O-COORDINATES IN CSPFE OF THE GATE UNDER
                                                                   G1 00290
        CCNSIDERATION.
                                                                   G1 00300
 IDXOE: LIST OF O-ERROR-COORDINATES IN CSPFE OF THE GATE UNDER
                                                                   G1 00310
        CONSIDERATION.
                                                                   G1 00320
  IDX1: LIST OF 1-COORDINATES IN CSPFE OF THE GATE UNDER
                                                                   G1 00330
        CCNSIDERATION.
                                                                   G1 00340
```

```
IDX1E: LIST OF 1-ERROR-COORDINATES IN CSPFE OF THE GATE UNDER
                                                                           G1 00350
C
              CONSIDERATION.
                                                                           G1 00360
C
       IFLAG: SAME AS EYEFLG IN SUBROUTINE PROCII.
                                                                           G1 00370
C
              INC$MX(GI,GJ)>0 MEANS THERE EXISTS A CONNECTION FROM GATE G1 00380
               (OR EX. VAR.) GI TO GATE GJ. INC$MX(GI,GJ)=0 IF NCT.
                                                                           G1 00390
C
      INPTCV: LISTS FOR EACH CORRESPONDING ENTRY OF F$1, HOW MANY INPUTSG1 00400
C
              HAVE A '1' IN THE POSITION INDICATED BY F$1.
                                                                           G1 00410
C
       IPATH: IPATH(GI)=1 MEANS GATE GI IS ON A PATH FROM A CERTAIN GATEG1 00420
              TO AN OUTPUT GATE. OTHERWISE IPATH(GI) = 0.
                                                                           G1 00430
       IPRED: IPRED(I,GJ) GIVES THE NAME OF THE I-TH GATE OR EX. VAR. ING1 00440
C
              A LIST OF GATES AND EX. VAR. FEEDING GJ.
                                                                           G1 00450
C
       I SUCC:
              ISUCC(I, GJ) GIVES THE NAME OF THE I-TH GATE FED BY GJ.
                                                                           G1 00460
C
       JFLAG: SAME AS JAYFLG IN SUBROUTINE PROCII.
                                                                           G1 00470
C
        KEYA: A FLAG INDICATING IF ANY ERROR COMPENSATION HAS BEEN
                                                                           G1 00480
C
              PERFORMED.
                                                                           G1 00490
        KEYB: A FLAG INDICATING IF ANY PRIMARY O-ERPOR-COORDINATES HAS
                                                                           G1 00500
              BEEN COMPENSATED.
                                                                           G1 00510
Ċ
       KFLAG: SAME AS KEIFLG IN PROCII.
                                                                           G1 00520
C
        LEVM: NUMBER OF LEVELS IN THE NETWORK (NOTE EX. VAR. ARE ALSO
                                                                           G1 00530
(
                                                                           G1 00540
              ASSIGNED LEVELS JUST LIKE GATES).
C
      LGLIST: LGLIST(J) TELLS NO. OF GATES AND EX. VAR. IN LEVEL J OF
                                                                           G1 00550
C
              NETWORK.
                                                                           G1 00560
Ç
         LIP: NUMBER OF PREDECESSORS FOR THE GATE UNDER CONSIDERATION.
                                                                           G1 00570
      LIPRED: LIPRED(GI) TELLS NO. OF IMMEDIATE PREDECESSORS OF GATE GI.G1 00580
       LISTC: OPDERED LIST OF CONNECTIBLE INPUTS TO GCO. ORDERED BY
                                                                           G1 00590
              DECREASING NO. OF O'S IN GCO COVERED.
                                                                           G1 00600
       LISTL: GROERED LIST OF GATES AND EX. VAR. WHICH GRIGINALLY FED
                                                                           G1 00610
              GCC AND WHICH HAVE NOT YET BEEN DISCONNECTED. ORDERED BY G1 00620
C,
              DECREASING NO. OF ESSENTIAL 1'S.
                                                                           G1 00630
O
      LISUCC: LISUCC(GI) TELLS NO. OF IMMEDIATE SUCCESSORS OF GATE (OF
                                                                           G1 00640
              EX. VAR.) GI.
                                                                           G1 00650
       LMTS2: UPPER LIMIT OF THE NUMBER OF ELEMENTS IN SET S2.
                                                                           G1 C0660
C
      LPOTAB: FOR GATE GI, LPOTAB(GI) POINTS TO LAST ROW OF PUTAB
                                                                           G1 00670
C
              CONCERNING GI.
                                                                           G1 00680
           M: NUMBER OF NETWORK DUTPUT GATES.
                                                                           G1 00690
           N: NUMBER OF EXTERNAL VARIABLES (OR INPUT FNC.) AVAILABLE.
                                                                           01 00700
000
              FOR ERROR COMPENSATION PROGRAMS. IF MORE THAN NEPMAX
                                                                           G1 00710
              FRROR POSITIONS OCCUR WHEN A PARTICULAR GATE IS REMOVED,
                                                                           G1 00720
C
              PROGRAM SKIPS ATTEMPT TO COMPENSATE FOR THAT GATE'S
                                                                           G1 00730
Ç
                       VALUE CAN BE SPECIFIED BY USER, OTHERWISE EQUAL G1 00740
              REMOVAL.
C
              TO ONE HALF OF N2 BY DEFAULT.
                                                                           G1 00750
                                                                           G1 00760
C
          NM: SUM OF N PLUS M
                                                                           G1 00770
              SUM OF NM PLUS 1.
         NM1:
                                                                           G1 00780
         NN2: PRODUCT OF N AND N2.
0
                                                                           G1 00790
         NOS: NUMBER OF ELEMENTS IN SET S.
                                                                           51 00800
        NCS1: NUMBER OF FLEMENTS IN SET S1.
C
      NOSISV: NUMBER OF ELEMENTS IN SET SI BEFORE ENTERING SUBROUTINE
                                                                           G1 00810
C
              RPLCF.
                                                                           G1 00820
C
                                                                           G1 00830
        NOS2: NUMBER OF ELEMENTS IN SET S2.
        NOT1: NUMBER OF ELEMENTS IN SET T1.
                                                                           G1 00840
C
      NOTISV: NUMBER OF ELEMENTS IN SET II BEFORE ENTERING SUBROUTINE
                                                                           G1 00850
C
              RPLCF.
                                                                           G1 00860
         NJO: NUMBER OF ACTIVE ELEMENTS IN ARRAY ICXO.
                                                                           G1 00870
                                                                           G1 00880
        NCOE: NUMBER OF ACTIVE ELEMENTS IN ARRAY IDXOE.
C
         NOI: NUMBER OF ACTIVE ELEMENTS IN ARRAY IDX1.
                                                                           G1 00890
        NOIE: NUMBER OF ACTIVE ELEMENTS IN ARRAY IDXIE.
                                                                           G1 00900
                                                                           G1 00910
1
          NR: SUM OF N PLUS R.
        NRM2: PRODUCT OF NR AND N2.
                                                                           G1 00920
                                                                           G1 00930
       NRPLC: NRPLC(I) STORES THE NUMBER OF ELEMENTS IN RPLC(I,*)
                                                                           G1 00940
                                                            FOR I=1,2.
          N1: SUM OF N PLUS 1.
                                                                           G1 00950
```

```
C
C
C
000
C
C
C
C
C
C
C
C
0000
000
```

LEVM

NRN2

```
N2: NUMBER OF DIFFERENT INPUT COMBINATIONS TO BE CONSIDERED
                                                                   G1 00960
        (USUALLY 2 TO THE POWER N).
                                                                   G1 00970
ORIGIN: ORIGIN(GI)=1 MEANS GI ORIGINALLY CONNECTED TO GCO.
                                                                   G1 00980
        ORIGIN(GI)= O MEANS GI DID NOT FEED GCO ORIGINALLY.
                                                                   G1 00990
    PS: PS(1.-) CONSECUTIVELY LISTS OUTPUTS OF EVERY EX. VAR. AND G1 01000
        EVERY GATE (FOR EVERY INPUT COMBINATION): P$(1,1),...,
                                                                   G1 01010
        P$(1,N2) FOR FIRST EX VAR; P$(1,N2+1), ..., P$(1,2*N2) FOR
                                                                   G1 01020
        SECOND EX VAR; ...; P$(1,N*N2+1),..., P$(1,N*N2+N2) FOR
                                                                   G1 01030
        FIRST GATE; ETC. P$(2,-) IS USED AS WORK SPACE FOR
                                                                   G1 01040
        CALCULATIONS ASSOCIATED WITH P$(1,-).
                                                                   G1 01050
   PCO: FOR ERROR COMPENSATION PROCEDURES. PCO IS THE GATE
                                                                   G1 01060
        REMOVED FROM CRIGINAL NETWORK TO OBTAIN CURRENT ALTERED
                                                                   G1 01070
        NETWORK.
                                                                   G1 01080
PDINTA: NOT USED.
                                                                   G1 01090
POINTC: POINTS TO LAST ELEMENT IN LISTC.
                                                                   G1 01100
PCINTL: POINTS TO LAST ELEMENT IN LISTL.
                                                                   G1 01110
POINTR: POINTS TO LAST ELEMENT IN RNEC1 (IN SUBROUTINE SUBSTI).
                                                                   G1 01120
POTAB: POSSIBLE OUTPUT TABLE. HOLDS INFORMATION ABOUT ALL
                                                                   G1 01130
        COMBINATIONS OF CONNECTIONS TO FORM NEW (AND HOPEFULLY
                                                                   G1 01140
        USEFUL) FUNCTIONS.
                                                                   G1 01150
PPOTAB: FOR GATE GI, PPOTAB(GI) POINTS TO FIRST OF A SEQUENCE OF
                                                                   G1 01160
        ROWS OF POTAB CONCERNING GI.
                                                                   G1 01170
        NUMBER OF GATES IN THE NETWORK (EXCLUDES EX VAR, ALSO -
                                                                   G1 01180
        NOTE SOME OF R GATES MAY BE ISOLATED).
                                                                   G1 01190
  RPLC: RPLC(1,*) STORES THE SELECTED GATE'S IP GATES WHICH HAVE
                                                                   G1 01200
                  ERROR-COORDINATES OF WEIGHT 2 OR ABOVE.
                                                                   G1 01210
        RPLC(2.*) STORES THE SELECTED GATE'S IP GATES WHICH HAVE
                                                                   G1 01220
                  AT LEAST ONE ERROR-COORDINATE OF WEIGHT 1.
                                                                   G1 01230
RSCONN: LIST OF CONNECTIONS ADDED TO A NETWORK (IN CODED FORM).
                                                                   G1 01240
PICCNN: LIST OF CONNECTIONS REMOVED FROM A NETWORK (CODED FORM).
                                                                   G1 01250
     S: NO. OF CONNECTIONS ADDED TO A NETWORK. POINTS TO LAST
                                                                   G1 01260
        ENTRY IN RSCONN.
                                                                   G1 01270
  SETS: SET S CONSISTING OF INPUTS OF THE GATE UNDER CONSIDERATIONG 1 01280
        WHICH ARE TO BE REPLACED IF POSSIBLE.
                                                                   G1 01290
 SETS1: SET S1 CONSISTING OF ELEMENTS OF SET S WHICH CAN BE
                                                                   G1 01300
        REPLACED BY ELEMENTS IN SET S2.
                                                                   G1 01310
 SETS2: SET S2 CONSISTING OF FUNCTIONS WHICH ARE CANDIDATES FOR
                                                                   G1 01320
        REPLACING ELEMENTS IN SET S.
                                                                   G1 01330
 SETT1: SET T1 CONSISTING OF ESSENTIAL ONES COVERED BY ELEMENTS ING1 01340
                                                         SET S1.
                                                                   G1 01350
   STS: STARTING ELEMENT OF SET S.
                                                                   G1 01360
SUC$MX: SUC$MX(GI,GJ)>O MEANS GATE GJ IS A SUCCESSOR OF GATE GI.
                                                                   G1 01370
                                                                   G1 01380
        SUC$MX(GI,GJ)=0 IF NOT.
  SUMP: SUM OF ALL ACTIVE INPUTS OF THE GATE UNDER CONSIDERATION. G1 01390
 SUMS2: SUM OF ALL ACTIVE ELEMENTS OF SET S2.
                                                                   G1 01400
     T: NUMBER OF CONNECTIONS REMOVED FROM A NETWORK. POINTS TO
                                                                   G1 01410
        LAST ENTRY IN RTCONN.
                                                                   G1 01420
  TIME: USED TO STORE AMOUNT OF ELAPSED COMPUTATION TIME.
                                                                   G1 01430
 UNAME: MNEMONIC NAMES FOR EXTERNAL VARIABLES AND GATES.
                                                                   G1 01440
VF$UB1: POINTS TO LAST ELEMENT IN VF$1.
                                                                   G1 01450
  VF$1: SIMILAR TC F$1, EXCEPT THIS LISTS JUST COMPONENT POSITIONSG1 01460
        (OF O'S IN CSPF VECTOR OF GCO) COVERED ONLY BY REMAINING
                                                                   G1 01470
        ORIGINALLY CONNECTED INPUTS TO GCO.
                                                                   G1 01480
                                                                   G1 01490
                                                                   G1 01500
                                                                   G1 01510
COMMON
       NEPMAX
                                                                   G1 01520
COMMON
                                                     , B
         N
                      , M
                                     , A
                                                                   G1 01530
                     , N2
                                     , N1
                                                     , NR
         R
     ,
                                                                   G1 01540
                     , KFLAG
                                     , JFLAG
         NM
                                                    , COST
                                                                   G1 01550
```

, NM1

, NN2

G1 01560

```
ISUCC(40,40) , LISUCC(40)
      COMMON
                                             , IPRED(40,40) , LIPRED(40)
                                                                              G1 01570
                INC$MX(40,40), SUC$MX(40,40), P$(2,1280)
                                                               • UNAME(40)
                                                                              G1 01580
           9
                            , LGLIST(40)
                                             , HLIST(40,40) , TIME
     2
                GLEVEL(40)
                                                                              G1 01590
                                               , 5
      COMMON
                              , RTCONN(100)
                                                               , RSCONN(100) G1 01600
                              .POINTA
      NOMMED
                IFLAG
                                               . ESSIS (40)
                                                               ,F$1(32)
                                                                              G1 01610
            ,F$UR1
                                               ,LISTC(40)
                                                               , POINTC
                              ,INPTCV(32)
     1
                                                                              G1 01620
                              , PCINTL
                                                               , IPATH(40)
     2
            ,LISTL(40)
                                               ORIGIN(40)
                                                                              G1 01630
            , POINTR
     3
                              , VF$1(32)
                                               .VF$UB1
                                                               .GSMALL(40, 32)G1 01640
      CCMMON
              POTAB(200,42),PPOTAB(40)
                                               , LPOTAB(40)
                                                               , NRPLC(2)
                                                                              G1 01650
           , RPLC (2,40)
                              ,IDX0(32)
                                               , IDX0E(32)
                                                               , IDX1(32)
                                                                              G1 01660
     1
     2
                                                                              G1 01670
            , ID X1E(32)
                                               ,SETT1(32)
                               ,SUMP(32)
                                                               , NOT1
                                               ,SETS(40)
                                                               , NOS
     3
            ,SETS1(40)
                              .NOS1
                                                                              G1 01680
                                                               .NCS2
            , STS
                              .SUMS 2 (32)
                                               , SETS2(200)
     4
                                                                              G1 01690
     5
                              NOOE
                                               , KEYA
                                                               . KEYB
                                                                              G1 01700
            , LIP
                              ,NO1
            , NO0
                                               .NO1E
                                                               , $GT
                                                                              G1 01710
                              .SPW
                                                               • GI
                                                                              G1 01720
           , $LTH
                                               . $NOE
                              NOT1SV
                                                              ,LMTS2
                                                                              G1 01730
      NOMMOD
                                              ,NOS1SV
      DIMENSION CHTLIS(144), UGATE(40), UHEAD(20)
                                                                              G1 01740
      DATA KOUNTS /O/. UBLANK/ 1/
                                                                              G1 01750
  990 PEAD(5,1000,END=500) UHEAD, N, M, R, A, B, UC, NEPMAX
                                                                              G1 01760
      NEPMAX IS THE MAXIMUM ALLOWABLE NUMBER OF ERROR POSITIONS
C
                                                                              G1 01770
 1000 FORMAT(20A4/514,A4,14)
                                                                              G1 01780
      KEYXC=0
                                                                              G1 01790
      IF (UC.NE. UBLANK) KEYXC =1
                                                                              G1 01800
      CALL PAGE
                                                                              G1 01810
                                                                              G1 01820
      CALL LINE(10)
      KOUNT5=KOUNT5+1
                                                                              G1 01830
                                                                              G1 01840
      PRINT 2, KOUNTS
    2 FORMAT (20X, **** OPTIMAL NOR NETWORK ****, 50X, *PROBLEM NO. = *, I4 ) G1 01850
      CALL LINE(4)
                                                                              G1 01860
      PRINT 1005, UHEAD
                                                                              G1 01870
 1005 FORMAT (25X, 20A4)
                                                                              G1 01880
      CALL LINE(4)
                                                                              G1 01890
      PRINT 10. N.M.A.B
                                                                              G1 01900
   10 FORMAT(30X, NUMBER OF VARIABLES = 1, 14 //
                                                                              G1 01910
     1
              30X, 'NUMBER OF FUNCTIONS =', 14 //
                                                                              G1 01920
              30X, COST COEFFICIENT A = 1, 14//
                                                                              G1 01930
     2
              47X.
                                     * B
                                         = 4 , [4]
                                                                              G1 01940
                                                                              G1 01950
      CALL LINE(1)
      IF(KEYXC.NE.O) GO TO 25
                                                                              G1 01960
      PRINT 21
                                                                              G1 01970
   21 FORMAT(1H0,29X,'--- UNCOMPLEMENTED VARIABLES X ---!)
                                                                              G1 01980
      GR TO 30
                                                                              G1 01990
   25 CONTINUE
                                                                              G1 02000
                                                                              G1 02010
      PRINT 28
   28 FORMAT(1H0,29X,'--- BOTH COMPLEMENTED AND UNCCMPLEMENTED VARIABLESGI 02020
     1 X, Y ---!)
                                                                              G1 02030
   30 CONTINUE
                                                                              G1 02040
      CALL LINE(5)
                                                                              G1 02050
C**** SET UP EXTERNAL VARIABLES ****
                                                                              G1 02060
                                                                              G1 02070
      N2 = 2 * * N
                                                                              G1 02080
      IF (NEPMAX.EQ.O) NEPMAX = N2/2
      H=N*N2
                                                                              G1 02090
                                                                              G1 02100
      J= N2
                                                                              G1 02110
      L= 1
                                                                              G1 02120
                                                                              G1 02130
      03 1011 II=1,N
                                                                              G1 02140
       J = J/2
       L=L#2
                                                                              G1 02150
                                                                              G1 02160
       SN= 1
                                                                              G1 02170
       00 1010 LE=1, L
```

```
SN=-SN
          V=(1+SN)/2
                                                                               G1 02180
          DO 1009 JJ=1,J
                                                                               G1 02190
                                                                               G1 02200
           I = I + 1
                                                                               G1 02210
           P$(1, I)=V
        IF (KEYXC.NE.O) P$ (1, I+H)=1-V
                                                                               G1 02220
  1009
                                                                               G1 02230
          CONTINUE
  1010 CENTINUE
                                                                               G1 02240
  1011 CONTINUE
                                                                               G1 02250
        IF(KEYXC.NE.O) N=N+N
                                                                               G1 02260
       N1 = N + 1
                                                                               G1 02270
                                                                               G1 02280
       NM=N+M
       NM1=NM+1
                                                                               G1 02290
       NN2=N*N2+1
                                                                               G1 02300
       NR=N+R
                                                                               G1 02310
                                                                               G1 02320
       NRN2=NR*N2
       CALL DUTPUT(INC $MX , KEYXC)
                                                                               G1 02330
 C***** READ IN NETWORK INFORMATION AND SET UP INC $MX *****
                                                                               G1 02340
       READ 1001,
                                                                               G1 02350
                     CNTLIS
  1001 FORMAT(1615)
                                                                               G1 02360
                                                                              G1 02370
       DO 1115 GI=1,NR
       DC 1115 GJ=1, NR
                                                                              G1 02380
  1115 INC $MX(GI,GJ) =0
                                                                              G1 02390
       DO 1120 I=1,144
                                                                              G1 02400
                                                                              G1 02410
        ITEM=CNTLIS(I)
       IF(ITEM.EQ.0) GO TO 1119
                                                                              G1 02420
        GI=ITEM/100
                                                                              G1 02430
        GJ=ITEM-100*GI
                                                                              G1 02440
       INC $MX(GI,GJ)=1
                                                                              G1 02450
       GO TO 1120
                                                                              G1 02460
 1119 COST=A*R+B*(I-1)
                                                                              G1 02470
                                                                              G1 02480
       GO TO 1130
 1120 CONTINUE
                                                                              G1 02490
 1130 CONTINUE
                                                                              G1 02500
      CALL SUBNET
                                                                              G1 02510
      CALL PVALUE
                                                                              G1 02520
      CALL LINE(4)
                                                                              G1 02530
      PRINT 1140, COST
                                                                             G1 02540
 1140 FORMAT (20X. ORIGINAL NETWORK
                                                                             G1 02550
                                         COST=1, 15)
      CALL LINE(4)
                                                                             G1 02560
      CALL TRUTH(P$,1)
                                                                             G1 02570
      CALL LINE(4)
                                                                             G1 02580
      CALL CKT(INC$MX,GLEVEL)
                                                                             G1 02590
                                                                             G1 02600
C**** ENTRY REDUNDANCY CHECK ****
                                                                             G1 02610
                                                                             G1 02620
      S = 0
      T = 0
                                                                             G1 02630
     CALL UNNECE
                                                                             G1 02640
     GATES = M
                                                                             G1 02650
     C = 0
                                                                             G1 02660
     DO 4 GI = 1,NR
                                                                             G1 02670
     C = C + LISUCC(GI)
                                                                             G1 02680
     IF(GI.LE.NM)GOTO4
                                                                             G1 02690
     IF(LISUCC(GI).GT.O)GATES=GATES+1
                                                                             G1 02700
   4 CONTINUE
                                                                             G1 02710
     OLDCST = A*GATES + B*(C)
                                                                             G1 02720
     T=0
                                                                            G1 02730
     S=0
                                                                            G1 02740
     INITIALIZE TIMER TO 10 MINUTES
                                                                            G1 02750
     CALL STIMEZ(60000)
                                                                            G1 02760
     TIME = KTIMEZ(O)
                                                                            G1 02770
                                                                            G1 02780
```

```
CALL PROCII(3,1,0,0)
                                                                             G1 02800
C
      CALL FOR ELAPSED TIME
                                                                             G1 02810
      TIME = KTIMEZ(0) - TIME
                                                                             G1 02820
      CALL LINE(4)
                                                                             G1 02830
      PRINT 3915
                                                                             G1 02840
 3916 FORMAT(20X, 'TIME ELAPSED = ', I8, ' CENTISECONDS')
                                                                             G1 02850
 3915 FORMAT (20X, 'NETWORK DERIVED BY PROCII')
                                                                             G1 02860
      PRINT 3916, TIME
                                                                             G1 02870
      CALL LINE(4)
                                                                             G1 02880
      CALL TRUTH(P$,1)
                                                                             G1 02890
      CALL LINE(4)
                                                                             G1 02900
      CALL CKT(INC $MX, GLE VEL)
                                                                             G1 02910
      GATES = M
                                                                             G1 02920
      C = 0
                                                                             G1 02930
      DO 36 GI = 1.NR
                                                                             G1 02940
      C = C + LISUCC(GI)
                                                                             G1 02950
                                                                             G1 02960
      IF(GI.LE.NM) GD TD 36
      IF(LISUCC(GI) \cdot GT \cdot O) GATES = GATES + 1
                                                                             G1 02970
   36 CONTINUE
                                                                             G1 02980
      NEWCST = A*GATES + B*C
                                                                             G1 02990
      IF (NEWCST.LT.OLDCST)GD TO 37
                                                                             G1 03000
      PRINT 105
                                                                             G1 03010
  105 FORMAT(1H ,10X, 'NO REDUNDANCY FOUND.')
                                                                             G1 03020
      GC TO 990
                                                                             G1 03030
   37 CALL LINE(3)
                                                                             G1 03040
      PRINT 320, NEWCST
                                                                             G1 03050
  320 FORMAT(9x, ** A NETWORK DERIVED BY PROCII*/9x, *COST=*, 15, *.*)
                                                                             G1 03060
      GJ TO 990
                                                                             G1 03070
  500 STOP
                                                                             G1 03080
      END
                                                                             G1 03090
      SUBROUTINE CONCODIGGO, CCO)
                                                                             G1 03100
C
      G1 03120
      THIS SUBROUTINE CONNECTS INPUT, CCO, CHOSEN IN BLOCK 25, TO
                                                                             G1 03130
         A GATE GCC. SOME ARRAYS ARE ALSO UPDATED.
                                                                             G1 03140
                                                                             G1 03150
         (CONCCO = CONNECT COD)
                                                                             G1 03160
                                                                             G1 03170
      IMPLICIT INTEGER * 4 (A-T, V-Z, $), REAL (U)
C
                                                                             G1 03180
C
      DEFINITIONS OF 'COMMON' VARIABLES CAN BE FOUND IN MAIN PROGRAM
                                                                             G1 03190
                                                                             G1 03200
      CTDWN, I, J, X, Y ARE JUST TEMPORARY VARIABLES USED IN THIS SUBROUTINEG 1 03210
                                                                             G1 03220
      COMMON NEPMAX
                                                                             G1 03230
      COMMON
               N
                              · M
                                                                В
                                                                             G1 03240
                                                Α
                R
                              , N2
                                                                             G1 03250
     1
                                                N1
                                                              , NR
            9
                              , KFLAG
                                                JFL AG
                                                                             G1 03260
     2
               NM
                                                                COST
            ,
                LEVM
                              · NRN2
                                                NM1
                                                                             G1 03270
     3
                                                                NN2
                                                IPRED(40,40)
                                                                             G1 03280
      COMMEN
                ISUCC(40,40)
                             , LISUCC(40)
                                                                LIPRED(40)
                                                              7
                                             .
                INC $MX(40,40), SUC $MX(40,40), P$(2,1280)
                                                              , UNAME (40)
                                                                             G1 03290
                              , LGLIST(40)
                                             , HLIST (40,40)
                GLEVEL (40)
                                                                TIME
                                                                             G1 03300
      NOMMOS
               T
                                                              , PSCCNN(100) G1 03310
                                              , 5
                              , RTCCNN(100)
                                                                             G1 03320
      COMMON
                IFLAG
                              , POINTA
                                              , ESS1S(40)
                                                              ,F$1(32)
                                                              , PO INTC
            ,F$UB1
                              ,INPTCV(32)
                                              LISTC(40)
                                                                             G1 03330
                                                                             G1 03340
            , LISTL(40)
                                                              , IPATH(40)
                              , PCINTL
                                              ORIGIN(40)
           . POINTR
                              , VF$1(32)
                                              , VF $UB1
                                                              ,GSMALL(40,32)G1 03350
                                              , LPOTAB(40)
                                                              , NRPLC(2)
                                                                             G1 03360
      COMMON
                POTAB(200,421, PPCTAB(40)
                             ,IDX0(32)
                                                                             G1 03370
           , RPLC (2, 40)
                                              , IDXOE(32)
                                                              ,IDX1(32)
```

C\*\*\*\* PROCEDURE NTCD (PROCII)

```
, SUMP (32)
                                               ,SETT1(32)
                                                                               G1 03380
            , IDX1E(32)
                                                               .NOT1
     3
                                                                               G1 03390
            , SETS1(40)
                              • NOS1
                                               .SETS(40)
                                                               • NOS
                                                                              G1 03400
     4
                              ,SUMS 2(32)
                                               , SETS2(200)
            · STS
                                                               NOS2
     5
            .LIP
                              , NOOE
                                               .KEYA
                                                               . KEYB
                                                                               G1 03410
                              ,NO1
            , NOO
                                               .NO1E
                                                               , $GT
     6
                                                                              G1 03420
            , $LTH
                              . SPW
                                               . SNOE
                                                               •GI
                                                                              G1 03430
                              NOT1SV
                                                              .LMTS2
      COMMON
                                              , NOSISV
                                                                              G1 03440
      X = (CCO-1)*N2
                                                                              G1 03450
      DO 11 J=1.F$UB1
                                                                              G1 03460
      I = F$1(J)
                                                                               G1 03470
      IF(P$(2.X+I).NE.1)GO TO 11
                                                                               G1 03480
      INPTCV(I) = INPTCV(I) + 1
                                                                              G1 03490
                                                                              G1 03500
   11 CONTINUE
      ENTRY CNCCO(GCO, CCO)
                                                                              G1 03510
      USE OF THIS ENTRY POINT BY-PASSES UPDATE OF INPTCV
                                                                              G1 03520
C
C
      UPCATE VF$1
                                                                              G1 03530
      X = (CCO-1) * N2
                                                                              G1 03540
      CTDWN = VF\$UB1 + 1
                                                                              G1 03550
    1 CTDWN = CTDWN - 1
                                                                              G1 03560
      IF(CTDWN.EQ.O) GO TO 2
                                                                              G1 03570
      IF(P$(2, VF$1(CTDWN)+X).EQ.O) GD TO 1
                                                                              G1 03580
      DO 3 I = CTDWN, VF$UB1
                                                                              G1 03590
    3 VF$1(I) = VF$1(I+1)
                                                                              G1 03600
      VF\$UB1 = VF\$UB1 - 1
                                                                              G1 03610
      GOTO1
                                                                              G1 03620
    2 CONTINUE
                                                                              G1 03630
C
      UPDATE INC $MX
                                                                              G1 03640
      INC$MX(CCO,GCO) = 1
                                                                              G1 03650
C
      UPDATE LISUCC AND ISUCC
                                                                              G1 03660
      X = LISUCC(CCO)
                                                                              G1 03670
      00 4 I=1,X
                                                                              G1 03680
      IF(ISUCC(I,CCO).GT.GCO)GO TO 5
                                                                              G1 03690
    4 CONTINUE
                                                                              G1 03700
                                                                              G1 03710
    5 CONTINUE
      LISUCC(CCO) = X + 1
                                                                              G1 03720
      DO 6 J=I,X
                                                                              G1 03730
      Y = X-J+I+1
                                                                              G1 03740
      ISUCC(Y,CCO) = ISUCC(Y-1,CCO)
                                                                              G1 03750
    6 CONTINUE
                                                                              G1 03760
      ISUCC(I,CCO) = GCO
                                                                              G1 03770
C
      UPDATE LIPRED, IPRED
                                                                              G1 03780
      X = LIPRED(GCO)
                                                                              G1 03790
      DO 8 I=1, X
                                                                              G1 03800
      IF(IPREC(I,GCO).GT.CCO)GO TO 9
                                                                              G1 03810
    8 CONTINUE
                                                                              G1 03820
      I = X + 1
                                                                              G1 03830
    9 \text{ LIPRED(GCO)} = X + 1
                                                                              G1 03840
      DO 10 J=I,X
                                                                              G1 03850
      Y = X - J + I + 1
                                                                              G1 03860
      IPRED(Y,GCO) = IPRED(Y-1,GCO)
                                                                              G1 03870
   10 CONTINUE
                                                                              G1 03880
  100 IPRED(I,GCO) = CCO
                                                                              G1 03890
C
      UPDATE SUCSMX
                                                                              G1 03900
      CALL SUCCES
                                                                              G1 03910
C
      UPCATE INPTCV
                                                                              G1 03920
      RETURN
                                                                              G1 03930
      END
                                                                              G1 03940
      SUBROUTINE ELANDO(GATE)
                                                                              G1 03950
```

C

```
"GATE" AND TO ORDER THE REST IN "LISTL" ACCORDING TO DECREAS-
                                                                          G1 03490
     ING NUMBER OF ESSENTIAL CNES.
                                                                          G1 04000
     (ELANDO = ELIMINATE AND ORDER)
                                                                          G1 04010
                                                                          G1 04020
  IMPLICIT INTEGER*4(A-T,V-Z,$), REAL(U)
                                                                          G1 04030
                                                                          G1 04040
  DEFINITIONS OF 'COMMON' VARIABLES CAN BE FOUND IN MAIN PROGRAM
                                                                          G1 04050
                                                                          G1 04060
  VARIABLE DEFINITIONS:
                                                                          G1 04070
    MAX1: MAXIMUM NUMBER OF ESSENTIAL 1'S BELONGING TO ANY
                                                                          G1 04080
           PREDECESSOR OF 'GATE'.
                                                                          G1 04090
   NUMIN: NUMBER OF INPUTS TO GATE 'GATE'.
                                                                          G1 04100
    SAVE: NAME OF A PREDECESSOR OF 'GATE'.
                                                                          G1 04110
                                                                          G1 04120
  CTDOWN, I, J, SAVE, X, Y, Z, XX, XY, YY ARE USED AS JUST TEMPORARY VARIABLEGI 04130
                                                                         SG1 04140
  COMMON NEPMAX
                                                                          G1 04150
                                           A
                                                                          G1 04160
  NEWMED
           N
                            M
                                                             В
           R
                            N2
                                                             NR
                                                                          G1 04170
                                            N1
       9
 2
           MI
                          . KFLAG
                                            JFLAG
                                                             COST
                                                                          G1 04180
                          · NRN2
 3
           LEVM
                                          , NM1
                                                           · NN2
                                                                          G1 04190
                                          , IPRED(40,40)
                                                           , LIPRED(40)
  NCMMDD
           ISUCC(40,40) , LISUCC(40)
                                                                          G1 04200
           INC $MX(40,40), SUC$MX(40,40), P$(2,1280)
                                                           , UNAME(40)
 1
       9
                                                                          G1 04210
                                          , HLIST (40, 40)
                                                          , TIME
                         , LGLIST(40)
 2
           GLEVEL (40)
                                                                          G1 04220
  COMMON
                          , RTCCNN(100)
                                          · S
                                                           , RSCONN(100) G1 04230
  NOWNED
            IFLAG
                          .POINTA
                                          , ESS1S(40)
                                                                          G1 04240
                                                           ,F$1(32)
       , F$UB1
                          , INPTCV(32)
                                          ,LISTC(40)
                                                           .PCINTC
                                                                          G1 04250
 1
                          , PCINTL
                                          ,ORIGIN(40)
 2
       ,LISTL(40)
                                                           , IPATH(40)
                                                                          G1 04260
 3
       , POINTR
                          , VF$1(32)
                                          . VF$UB1
                                                           GSMALL(40,32)G1 04270
  COMMON
          POTA3(200,42),PPCTAB(40)
                                          , LPOTAB (40)
                                                           ,NRPLC(2)
                                                                          G1 04280
                          ,IDX0(32)
       , RPLC (2, 40)
                                          , IDX0E(32)
                                                           , IDX1(32)
                                                                          G1 04290
 1
 2
       , IDX1E(32)
                          ,SUMP (32)
                                          ,SETT1(32)
                                                           ,NOT1
                                                                          G1 04300
                                          , SETS (40)
 3
       , SETS1(40)
                          ,NOS1
                                                           , NOS
                                                                          G1 04310
                                          , SETS2 (200)
       , STS
                          .SUMS2(32)
                                                           .NOS2
                                                                          G1 04320
       .LIP
 5
                          , NOOE
                                          ,KEYA
                                                                          G1 04330
                                                           , KEYB
       , NCO
                                          .NOIE
                                                           , $GT
 6
                          ,NO1
                                                                          G1 04340
                          , SPW
       , SLTH
                                          . SNOE
                                                           •GI
                                                                          G1 04350
  COMMON
                          N JT 1SV
                                         , NOSISV
                                                          ,LMTS2
                                                                          G1 04360
                                                                          G1 04370
  ELIMINATE NON-ESSEN. INPUTS FIRST
  Y = 0
                                                                          G1 04380
  X = 0
                                                                          G1 04390
  NUMIN = LIPRED(GATE)
                                                                          G1 04400
  IF(IPATH(GATE).EQ.O.DR.KFLAG.NE.2) GC TO 1
                                                                          G1 04410
  SPECIAL INSTRUCTIONS FOR PROCIV (I. E., KFLAG = 2): PREFER
                                                                          G1 04420
                                                                          G1 04430
  CONNECTIONS FROM SUBNETWORK S(GCO) FOR FIRST REMOVAL IF GATE
  IS IN S.
                                                                          G1 04440
9 NUMIN = LIPRED(GATE)
                                                                          G1 04450
  NIMUN, I = I OI COI
                                                                          G1 04460
  SAVE = IPRED(I,GATE)
                                                                          G1 04470
  IF(IPATH(SAVE).EQ.O) GO TO 10
                                                                          G1 04480
 IF (ORIGIN(SAVE).EQ.O) GO TO 10
                                                                          G1 04490
                                                                          G1 04500
  IF(ESSIS(SAVE).NE.O) GO TO 10
 CALL RNONES (GATE, SAVE, 1)
                                                                          G1 04510
  Z = (SAVE-1)*N2
                                                                          G1 04520
                                                                          G1 04530
  0313 I=1,F$UR1
  IF (INPTCV(F$1(I)).NE.1)GC TO13
                                                                          G1 04540
 IF(P$(2,Z+F$1(I)).NE.1)GG T013
                                                                          G1 04550
                                                                          G1 04560
 XX = LIPRED(GATE)
                                                                          G1 04570
 D014 J=1, XX
```

THIS SUBROUTINE IS TO ELIMINATE NON-ESSENTIAL INPUTS TO GATE

G1 03970

G1 03980

```
G1 04580
   XY = IPRED(J,GATE)
                                                                         G1 04590
   YY = (XY-1)*N2
                                                                         G1 04600
   IF(P$(2.YY+F$1(I)).NE.1)GO TO14
   ESSIS(XY) = ESSIS(XY)+1
                                                                         G1 04610
   GO T013
                                                                         G1 04620
                                                                         G1 04630
14 CONTINUE
                                                                         G1 04640
13 CONTINUE
                                                                         G1 04650
   GO TO 9
                                                                         G1 04660
10 CONTINUE
 1 IF(X.EQ.NUMIN)GO TO 2
                                                                         G1 04670
   X = X + 1
                                                                         G1 04680
   Y = Y + 1
                                                                         G1 04690
   SAVE = IPRED(Y, GATE)
                                                                         G1 04700
   IF (OR IGIN( SAVE) . EQ. 0) GOTO1
                                                                         G1 04710
   IF(ESSIS(SAVE).NE.O) GO TO 1
                                                                         G1 04720
   IF HERE.
                 " SAVE " IS A NON-ESSEN. INPUT TO GATE, SO REMOVE
                                                                         G1 04730
   CALL RNCNES(GATE, SAVE,
                                 1)
                                                                         G1 04740
   Y = Y - 1
                                                                         G1 04750
   FINISH UPDATE OF ESSIS (RNONES DOES NOT COMPLETELY UPDATE)
                                                                         G1 04760
                                                                         G1 04770
   Z = (SAVE-1)*N2
   DO 3 I=1.F$UB1
                                                                         G1 04780
   IF(INPTCV(F$1(I)).NE.1)GO TO 3
                                                                         G1 04790
   IF(P$(2,Z+F$1(I)).NE.1)GO TO 3
                                                                         G1 04800
   XX = LIPRED(GATE)
                                                                         G1 04810
                                                                         G1 04820
   DO 4 J=1,XX
   XY = IPRED(J,GATE)
                                                                         G1 04830
   YY = (XY-1)*N2
                                                                         G1 04840
   IF(P$(2, YY+F$1(I)).NE.1)GC TO 4
                                                                         G1 04850
   ESSIS(XY) = ESSIS(XY)+1
                                                                         S1 04860
  GO TO 3
                                                                         G1 04870
4 CONTINUE
                                                                         G1 04880
3 CONTINUE
                                                                         G1 04890
  GO TO 1
                                                                         G1 04900
 2 CONTINUE
                                                                         G1 04910
   ENTRY ORDERLIGATE)
                                                                         G1 04920
   IF (KFLAG. EQ. 3) RETURN
                                                                         G1 04930
  NOW ORDER REMAINING INPUTS IN "LISTL"
                                                                         G1 04940
   POINTL = 0
                                                                         G1 04950
   MAX1 = 0
                                                                         G1 04960
     FIRST FIND MAX NUMBER OF ESSENTIAL ONES
                                                                         G1 04970
   X = LIPRED(GATE)
                                                                         G1 04980
  DO 5 I=1,X
                                                                         G1 04990
   IF(ORIGIN(IPRED(I,GATE)).EQ.0)GO TO 5
                                                                         G1 05000
  Y= ESS1S(IPRED(I,GATE))
                                                                         G1 05010
   IF(Y.LE.MAX1)GO TO 5
                                                                         G1 05020
  MAX1 = Y
                                                                         G1 05030
5 CONTINUE
                                                                         G1 05040
   IF(MAX1.EQ.O)GOTO8
                                                                         G1 05050
  CTDOWN = MAX1 + 1
                                                                         G1 05060
    NOW SEARCH FOR INPUTS WITH SUCCESSIVELY SMALLER NUMBERS
                                                                         G1 05070
    OF ESSENTIAL ONES
                                                                         G1 05080
  DO 6 I=1, MAX1
                                                                         G1 05090
  CTDOWN = CTDOWN - 1
                                                                         G1 05100
  DO 7 J=1, X
                                                                         G1 05110
   IF(ORIGIN(IPREC(J, GATE)).EQ.0)GO TO 7
                                                                         G1 05120
   Y = ESSIS(IPRED(J.GATE))
                                                                         G1 05130
   IF(Y.NE.CTDOWN)GO TO 7
                                                                         G1 05140
  PCINTL = POINTL + 1
                                                                         G1 05150
   LISTL(PCINTL) = IPREC(J, GATE)
                                                                         G1 05160
 7 CONTINUE
                                                                         G1 05170
6 CONTINUE
                                                                         G1 05180
```

C

C

C

```
SUBROUTINE MINI2(IMPROV)
                                                                    G1 05220
 THE NAME ATTEMPTS TO INDICATE THAT THIS SUBROUTINE IS A MINIATURE G1 05240
 VERSION OF PROCEDURE II (PROCII) - ACTUALLY, THIS ROUTINE GNLY
                                                                    G1 05250
 REMOVES CONNECTIONS, NONE ARE ADDED
                                                                    G1 05260
 IMPLICIT INTEGER*4(A-T,V-Z,$), REAL(U)
                                                                    S1 05270
                                                                    G1 05280
DEFINITIONS OF "COMMON" VARIABLES CAN BE FOUND IN MAIN PROGRAM.
                                                                    G1 05290
                                                                    G1 05300
 VARIABLE DEFINITIONS:
                                                                    G1 05310
 BESTSL: NAME OF A PRIORITY CANDIDATE TO DISCONNECT FROM GATE GCO. G1 05320
 CHOICE: NAME OF A GATE CHOSEN TO BECOME A COVER.
                                                                    G1 05330
 COMPNT: A COMPONENT OF AN INTERMEDIATE CSPF VECTOR.
                                                                    G1 05340
 EFLAG: SIGNALS WHICH ENTRY POINT USED.
                                                                    G1 05350
 FFEDGT: A GATE FEEDING GATE 'GATE'.
                                                                    G1 05360
 F$UBO: NUMBER OF "NECESSARY ZEROS" LISTED IN F$O.
                                                                    G1 05370
    F&O: LISTS (CONSECUTIVELY) POSITIONS OF NECESSARY ZEROS IN A
                                                                    G1 05380
         CONNECTABLE FUNCTION VECTOR.
                                                                    G1 05390
  GATE: NAME OF A GATE.
                                                                    G1 05400
GCOUNT: A COUNTER.
                                                                    G1 05410
 GORDER: A SPECIAL ORDERING OF GATES AND EXTERNAL VARIABLES SUCH
                                                                    G1 05420
         THAT NO GATE SUCCEEDS A PREDECESSOR IN THE ORDERING.
                                                                    G1 05430
MARKED: MARKED(GI)=1 FOR GI FEEDING 'GATE' INDICATES THAT GI HAS
                                                                    G1 05440
         ALREADY BEEN ASSIGNED NECESSARY ZEROS CORRESPONDING TO
                                                                    G1 05450
         "1" COMPONENTS IN THE CSPF VECTOR FOR "GATE".
                                                                    G1 05460
NMINLY: NUMBER OF GATES IN A CERTAIN LEVEL OF THE NETWORK.
                                                                    G1 05470
 SELECT: NAME OF AN INPUT SELECTED AS A CANDIDATE FOR DISCONNECTIONS 1 05480
         FROM GATE 'GCO'.
                                                                    G1 05490
      T: COUNTS REMOVED CONNECTIONS.
                                                                     G1 05500
 TURDER: A SPECIAL ORDERING OF GATES AND EXTERNAL VARIABLES SUCH
                                                                     G1 05510
         THAT VARIABLES COME FIRST FOLLOWED BY GATES WITH DECREASEDG1 05520
         NUMBERS OF OUTPUTS (TIES ARE BROKEN BY GORDER).
                                                                    G1 05530
 TPOINT:
         POINTER TO TORDER.
                                                                    G1 05540
 TIPRED: LIST OF GCO'S PREDECESSORS AT ONE STAGE OF COMPUTATION.
                                                                    G1 05550
 T2PFED: LIST OF GCO'S PREDECESSORS AT ONE STAGE OF COMPUTATION.
                                                                    G1 05560
 TISUB: A POINTER TO TIPRED.
                                                                    G1 05570
 T2SUB: A POINTER TO T2PRED.
                                                                    31 05580
  USED: USED(GI)=1 MEANS GI IS AN DUTPUT GATE, OR IS A COVER FOR
                                                                    G1 05590
         SOME O-COMPONENT OF 'GATE'. (IT ALSO HAS A TEMPORARY USE
                                                                    G1 05600
         IN BEGINNING OF PROGRAM.)
                                                                    G1 05610
                                                                    G1 05620
COUNT, I, II, J, K, L, MOST, Q, TCOUNT, X, XX, Y ARE USED AS JUST TEMPORARY G1 05630
                                                          VARIABLES.G1 05640
HOW TO INCREASE CAPACITY OF SUBROUTINE.
                                                                    G1 05650
DIMENSION: TIPRED(X), T2PRED(X), GCRDER(X),
                                                                    G1 05660
                                            - X EQUAL TO MAX NUMBER G1 05670
            MARKED(X), USED(X), TORDER(X)
                                              OF GATES PLUS EXTERNALGI 05680
                                              VARIABLES.
                                                                    G1 05690
            F$O(Y) - Y EQUAL TO: 2**(MAX ALLOWED NO. OF EX. VAR.)
                                                                    G1 05700
                                                                    G1 05710
COMMON NEPMAX
                                                                    G1 05720
                                                                    G1 05730
COMMON
          N
                         M
                                                        В
                                         Α
          R
                                                                    G1 05740
1
                         N2
                                         N1
                                                        NR
                         KFLAG
2
          Nº M
                                         JFLAG
                                                        COST
                                                                    G1 05750
                                                                    G1 05760
          LEVM
                         NR N2
                                        NM1
3
                                                        NN2
COMMON
                                                                    G1 05770
          ISUCC(40,40) , LISUCC(40)
                                        IPRED(40,40) , LIPRED(40)
```

```
INC$MX(40,40), SUC$MX(40,40), P$(2,1280)
                                                               , UNAME(40)
                                                                              G1 05780
                                                                              G1 05790
                                              , HLIST (40,40)
                GLEVEL (40)
                            , LGLIST(40)
                                                               , TIME
                                               , 5
                                                               , RSCONN(100) G1 05800
                              , RTCONN(100)
      COMMON
                Т
                              .POINTA
                                               , ESS1S(40)
                                                               ,F$1(32)
      COMMON
                IFLAG
                                                                              G1 05810
                                               ,LISTC(40)
                                                               , POINTC
           ,F$UB1
                              ,INPTCV(32)
                                                                              G1 05820
                              , POINTL
                                               ORIGIN(40)
                                                               , IPATH(40)
                                                                              G1 05830
           .LISTL (40)
     2
                              , VF$1(32)
           , POINTR
     3
                                               ,VF$UB1
                                                               , GSMALL (40, 32) G1 05840
                                                               ,NRPLC(2)
                POTAB(200,42), PPOTAB(40)
                                               ,LPOTAB(40)
                                                                              G1 05850
      COMMON
                                                                              G1 05860
                                               , IDXOE(32)
                                                               , IDX1(32)
           ,RPLC(2,40) ,IDXO(32)
           . IDX1E(32)
                              .SUMP(32)
                                               , SETT1(32)
                                                               , NOT1
                                                                              G1 05870
     2
                                               , SETS (40)
     3
           , SETS1(40)
                              , NOS1
                                                               , NOS
                                                                              G1 05880
                              ,SUMS2(32)
                                               , SETS2 (200)
                                                               , NOS2
                                                                              G1 05890
           , STS
                                               , KEYA
                                                                              G1 05900
     5
                                                               , KEYB
           , LIP
                              NOOE
                                                                              G1 05910
                              , NO1
                                               ,NOIE
                                                               , $GT
     6
           OCH.
                              ,SPW
                                              , $NOE
                                                               , GI
                                                                              G1 05920
           . $LTH
                                                              .LMTS2
      COMMON
                              NOT1SV
                                              , NOS 1SV
                                                                              G1 05930
      DIMENSION T1PRED(40), T2PRED(40), GORDER(40), F$0(32), MARKED(40)
                                                                              G1 05940
                                                                              G1 05950
      DIMENSION USED(40), TORDER(40)
      IMPROV = 0
                                                                              G1 05960
                                                                              G1 05970
      T = 0
      OPDER GATES IN GORDER
                                                                              G1 05980
C
      EFLAG = 0
                                                                              G1 05990
                                                                              G1 06000
      GO TO 63
      THIS ENTRY POINT FOR CALCULATION OF GORDER ONLY
                                                                              G1 06010
      ENTRY FORMGO
                                                                              G1 06020
      EFLAG = 1
                                                                              G1 06030
   63 CONTINUE
                                                                              G1 06040
                                                                              G1 06050
      COUNT = 0
                                                                              G1 06060
      DO 1 I=1, LEVM
                                                                              G1 06070
      NMINLV = LGLIST(I)
      IF (NMINLV. EQ.O)GOTO1
                                                                              G1 06080
                                                                              G1 06090
      DO 2 J=1.NMINLV
                                                                              G1 06100
      COUNT = COUNT + 1
                                                                              G1 06110
      GORDER (COUNT) = HLIST (J.I)
    2 CONTINUE
                                                                              G1 06120
    1 CONTINUE
                                                                              G1 06130
      IF (EFLAG.EQ.1) RETURN
                                                                              G1 06140
C
      CALCULATE NUMBER OF OUTPUTS OF EACH GATE
                                                                              G1 06150
      (THE ARRAY 'USED' IS USED HERE JUST TEMPORARILY)
                                                                              G1 06160
      DO 51 I=N1,NR
                                                                              G1 06170
      TCOUNT = 0
                                                                              G1 06180
      DO 52 J=1.NR
                                                                              G1 06190
                                                                              G1 06200
      IF(INC\$MX(I,J).EQ.1)TCOUNT = TCOUNT + 1
   52 CONTINUE
                                                                              G1 06210
C
      TCOUNT NOW CONTAINS THE NUMBER OF OUTPUTS OF GATE I
                                                                              G1 06220
      USED(I) = TCOUNT
                                                                              G1 06230
                                                                              G1 06240
   51 CONTINUE
      MOST = 0
                                                                              G1 06250
      DO 53 I =N1,NR
                                                                              G1 06260
      IF(USED(I).GT.MCST)MOST = USED(I)
                                                                              G1 06270
   53 CONTINUE
                                                                              G1 06280
      DC 56 I = 1.N
                                                                              G1 06290
                                                                              G1 06300
   56 \text{ TORDER(I)} = I
      TPCINT = N1
                                                                              G1 06310
      MOST = MOST + 1
                                                                              G1 06320
   50 \text{ MOST} = \text{MOST} - 1
                                                                              G1 06330
      IF (MOST.LT.O)GO TO 54
                                                                              G1 06340
      DO 55 I=1,NR
                                                                              G1 06350
      II = GORDER(I)
                                                                              G1 06360
      IF(II.LE.N)GO TO 55
                                                                              G1 06370
      IF(USED(II).NE.MOST)GO TO 55
                                                                              G1 06380
```

```
TORDER (TPOINT) = II
                                                                               G1 06390
      TPOINT = TPOINT + 1
                                                                               G1 06400
   55 CONTINUE
                                                                               G1 06410
                                                                               G1 06420
      GO TO 50
   54 CONTINUE
                                                                               G1 06430
      INITIALIZE GSMALL
                                                                               G1 06440
      DO 4 I=N1,NM
                                                                               G1 06450
      X = (I-1)*N2
                                                                               G1 06460
      DO 4 J=1,N2
                                                                               G1 06470
      Y = P$(1,X+J)
                                                                               G1 06480
      IF(Y.EQ.O)GSMALL(I,J) = -100
                                                                               G1 06490
      IF(Y.EQ.1)GSMALL(I,J) = 1
                                                                               G1 06500
      IF(Y.EQ.-1)GSMALL(I.J)=0
                                                                               G1 06510
    4 CONTINUE
                                                                               G1 06520
      EFLAG = 0
                                                                               G1 06530
      GD TO 57
                                                                               G1 06540
      ENTRY INITGS
                                                                               G1 06550
      EFLAG = 1
                                                                               G1 06560
   57 00 3 I=1,NR
                                                                               G1 06570
      USED(I) = 0
                                                                               G1 06580
      IF (I.LT.N1)GD TD 58
                                                                               G1 06590
      IF(I.GT.NM) GO TO 58
                                                                               G1 06600
      G0 T0 3
                                                                               G1 06610
   58 D3 59 J = 1.N2
                                                                               G1 06620
   59 GSMALL(I,J)= 0
                                                                               G1 06630
    3 CONTINUE
                                                                               G1 06640
      DO 62 I = N1.NM
                                                                               G1 06650
      USED(I) = 1
                                                                               31 06660
   62 SONTINUE
                                                                               G1 06670
C
      INITIALIZATION
                                                                               G1 06680
      DO 34 I=1.NR
                                                                               G1 06690
      GATE = GORDER(I)
                                                                               G1 06700
                                                                               G1 06710
      IF(SATE-LT-N1)GD TO 34
      XX= LIPKED (GATE)
                                                                               31 06720
      IF(XX.EQ.O)GOTO34
                                                                               G1 06730
      F$UB1 = 0
                                                                               G1 06740
      F$UBO = 0
                                                                               G1 06750
      00 35 J=1.N2
                                                                               G1 06760
      COMPNT = GSMALL(GATE, J)
                                                                               G1 06770
      IF(COMPNT.EQ.O)GO TO 35
                                                                               G1 06780
                                                                               G1 06790
      IF(COMPRT.LT.O)GD TO 36
      IF(COMPNT.GE.1000) GO TO 35
                                                                               G1 06800
      F$UBO = F$UBO + 1
                                                                               G1 06810
      FSC(FSUBO) = J
                                                                               31 06820
      GO TO 35
                                                                               G1 06830
   36 IF(COMPNT.LE.-1000) GO TO 35
                                                                               G1 06840
                                                                               G1 06850
      FSUR1 = FSUR1 + 1
      F$1(F$UB1) = J
                                                                               G1 06860
   35 CONTINUE
                                                                               G1 06870
      IF(F$UB1.EQ.0)GO TO 34
                                                                               G1 06880
      DD 38 K=1,XX
                                                                               G1 06890
      FEEDGT = IPRED(K.GATE)
                                                                               G1 06900
                                                                               G1 06910
      X = \{FEFDGT-1\}*N2
                                                                               G1 06920
      ng 39 L=1,F$UR1
      Y = F$1(L)
                                                                               G1 06930
      IF(P$(1, X+Y).LE.0)GD TO 39
                                                                               G1 06940
      IF(GSMALL(FEEDGT, Y).GT.1000)GOT039
                                                                               G1 06950
      IF(GSMALL(GATE,Y).EQ.-200)GOT039
                                                                               G1 06960
                                                                               G1 06970
      IF(GSMALL(GATE,Y).EQ.-100)GO TO 40
                                                                               G1 06980
      GSMALL(GATE,Y) = -200
      G7 T0 39
                                                                               G1 06990
```

```
40 GSMALL (GATE, Y) = -FEEDGT
                                                                          G1 07000
                                                                          G1 07010
39 CONTINUE
                                                                          G1 07020
38 CONTINUE
                                                                          G1 07030
   00 60 K=1.XX
                                                                          G1 07040
60 MARKED(IPRED(K, GATE)) = 0
   DO 41 K=1,F$UB1
                                                                          G1 07050
                                                                          G1 07060
   X = GSMALL(GATE, F$1(K))
                                                                          G1 07070
   IF (X.EQ.-100)GD TO 41
                                                                          G1 07080
   IF(X.EQ.-200)GOT041
                                                                          G1 07090
   X = -X
   GS MALL (+X, F$1 (K))=1
                                                                         G1 07100
   USED(X) = 1
                                                                          G1 07110
   IF (MARKED(X).EQ.1)GOTO41
                                                                          G1 07120
                                                                          G1 07130
   MARKED(X) = 1
   DO 42 L=1,F$UBO
                                                                          G1 07140
                                                                          G1 07150
   Y = GSMALL(X,F\$O(L))
   IF(Y.GT.1000.DR.Y.LT.-1000)GO TO 42
                                                                         G1 07160
   GSMALL(+X,F$0(L))=-100
                                                                          G1 07170
                                                                         G1 07180
42 CONTINUE
                                                                          G1 07190
41 CONTINUE
                                                                         G1 07200
34 CONTINUE
                                                                         G1 07210
   IF (EFLAG. EQ. 1) RETURN
   INITIALIZE COUNTER TO LOOP ONCE FOR EACH GATE
                                                                          G1 07220
                                                                          G1 07230
   GCOUNT = 0
   INCREMENT GCOUNT
                                                                         G1 07240
 5 GCOUNT = GCOUNT + 1
                                                                         G1 07250
   ARE ALL GATES EXHAUSTED?
                                                                          G1 07260
                                                                          31 07270
   IF (GCOUNT.LE.NR)GO TO 6
   IF(T.GT.O) IMPROV = 1
                                                                          G1 07280
                                                                         G1 07290
   IF (IMPROV. EQ.O)RETURN
   IF HERE, NETWORK WAS ALTERED, SO UPDATE ARRAYS
                                                                         G1 07300
  CALL SUBNET
                                                                          G1 07310
  CALL PVALUE
                                                                         G1 07320
  RETURN
                                                                         G1 07330
6 GCO = GORDER(GCOUNT)
                                                                         G1 07340
   IS GCO AN ISOLATED GATE OR EXTERNAL VARIABLE?
                                                                         G1 07350
   IF(GCO.LE.N)GOTO5
                                                                          G1 07360
  DO 8 I=1,N2
                                                                         G1 07370
   IF (GS MALL(GCO. I).GE.1)GOTO7
                                                                         G1 07380
 8 CONTINUE
                                                                         G1 07390
   IF HERE, GATE IS ISOLATED - REMOVE INPUTS
                                                                         G1 07400
   X = LIPRED(GCC)
                                                                         G1 07410
   IF(X.EQ.0)GOTO5
                                                                         G1 07420
   DO 9 I=1,X
                                                                         G1 07430
   Y = IPRED(I,GCO)
                                                                         G1 07440
   INC$MX(Y,GCO) = 0
                                                                         G1 07450
  RECORD THE DISCONNECTION
                                                                         G1 07460
   T = T + 1
                                                                         G1 07470
9 CONTINUE
                                                                         G1 07480
  GOTO 5
                                                                         G1 07490
   REMOVE UNNECESSARY CONNECTIONS TO GCO IN THE NEXT FEW SECTIONS
                                                                         G1 07500
                                                                         G1 07510
  CALCULATE F(GCO)
                                                                         G1 07520
 7 F$UB1 = 0
                                                                         G1 07530
  DO 10 I=1, N2
                                                                         G1 07540
   IF (GS MALL (GCO. I).GE.O)GOTO10
                                                                         G1 07550
   F$UB1 = F$UB1 + 1
                                                                         G1 07560
   F$1(F$UB1) = I
                                                                         G1 07570
10 CONTINUE
                                                                         G1 07580
  DO 11 I=1.F$UB1
                                                                         G1 07590
                                                                         G1 07600
11 INPTCV(F$1(I)) = 0
```

C

C

C

C

C

C

C

C

```
X = LIPRED(GCO)
                                                                               G1 07610
      DO 222 I = 1, X
                                                                               G1 07620
                                                                               G1 07630
      ESSIS(IPRED(I,GCO)) = 0
  222 CONTINUE
                                                                               G1 07640
      T1SUB = 0
                                                                              G1 07650
      T2SU9 = 0
                                                                               G1 07660
      DC 48 I = 1.NR
                                                                               G1 07670
      IF(INC$MX(I,GCO).EQ.O)GOTO48
                                                                               G1 07680
      T1SU9 = T1SUB + 1
                                                                               G1 07690
      TIPRED(TISUB) = I
                                                                               G1 07700
   48 CONTINUE
                                                                               G1 07710
   17 DO 18 I=1.X
                                                                               G1 07720
      Y = (T1PRED(I)-1)*N2
                                                                              G1 07730
      00 19 J=1,F$UB1
                                                                               G1 07740
      Q = F$1(J)
                                                                               G1 07750
      IF(P$(1,Y+Q).NE.1)GO TO 19
                                                                               G1 07766
      I=(INPTCV(Q).LE.O) GO TO 20
                                                                               G1 07770
      INPTCV(Q) = INPTCV(Q) + 1
                                                                               G1 07780
      GO TO 19
                                                                               G1 07790
   20 IF(INPTCV(Q).LT.0)G0 TO 21
                                                                               G1 07800
      INPTCV(Q) = -TIPRED(I)
                                                                              G1 07810
      GO TO 19
                                                                               G1 07820
   21 \text{ INPTCV(Q)} = 2
                                                                               31 07830
                                                                              G1 07840
   19 CONTINUE
   18 CENTINUE
                                                                               G1 07850
      MARK ESSENTIAL 1'S
                                                                               G1 07860
      DO 22 [=1,F$UB1
                                                                               G1 07370
      Q = INPTCV(F$1(I))
                                                                               G1 07880
      IF(Q.GE.0)G3 TC 22
                                                                               G1 07890
      ESSIS(-Q) = ESSIS(-Q) + 1
                                                                               G1 07900
                                                                               G1 07910
   22 CONTINUE
   46 SELECT = 0
                                                                               G1 07920
      BESTSL = 0
                                                                               G1 07930
      D7 45 L=1,X
                                                                               G1 07940
      Q = T1PRED(L)
                                                                               G1 07950
      IF (INC $ MX (Q, GCO) . EQ.O) GOTO 45
                                                                               G1 07960
      !F(ESS1S(0).GT.0)G0T045
                                                                               G1 07970
      IF(SELECT.EQ.O)SELECT = Q
                                                                               G1 07980
      IF(USED(Q).EQ.1)GCTO45
                                                                               G1 07990
      IF(BESTSL.NE.O)G9T045
                                                                               G1 08J00
      BESTSL = Q
                                                                               G1 08010
   45 CONTINUE
                                                                               G1 08020
      IF (SELECT.EQ.O)GD TO 47
                                                                               G1 08030
                                                                               G1 08040
      9 = SELECT
      IF(BESTSL.NE.O)Q = BESTSL
                                                                               G1 08050
      IF HERE, GATE HAS NO ESSENTIAL 1'S - REMOVE IT
                                                                               G1 08060
      IAC$MX(Q,GCG) = 0
                                                                               G1 08070
      T = T + 1
                                                                               G1 08080
      UPDATE ESSIS
C
                                                                               G1 08090
      Y = (Q - 1) * N2
                                                                               G1 08100
                                                                               G1 08110
      DO 24 J=1,F$UB1
                                                                               G1 08120
      V = F$1(J)
      IF (P$(1,Y+V).NE.1)GO TO 24
                                                                               G1 08130
      UPDATE INPTOV FOR COMPONENT V
                                                                               G1 08140
0
      INPTCV(V) = INPTCV(V) - 1
                                                                               G1 08150
                                                                               G1 08160
      IF (INPTC V(V).GT.11GO TO 24
      CASE WHEN NEW ESSEN 1 CREATED
                                                                               G1 08170
C
      D9 27 K = 1.X
                                                                               G1 08180
                                                                               G1 08190
      W = TIPRED(K)
                                                                              G1 08200
      IF(INC$MX(W,GCO).EQ.O) GO TO 27
                                                                               G1 08210
      Z = \{W - 1\} * N2
```

```
IF(P$(1,Z+V).EQ.0)GO TO 27
                                                                              G1 08220
                                                                              G1 08230
      ESSIS(W) = ESSIS(W) + 1
      IN THIS CASE, NO NEED TO UPDATE INPTCV FURTHER
                                                                              G1 08240
C
      GSMALL(GCO,V) = -W
                                                                              G1 08250
      GO TO 24
                                                                              G1 08260
                                                                              G1 08270
   27 CONTINUE
   24 CONTINUE
                                                                              G1 08280
      GOTO46
                                                                              G1 08290
   47 DO 49 I = 1.NR
                                                                              G1 08300
      IF(INC$MX(I,GCO).EQ.O)GOTO49
                                                                              G1 08310
      T2SUB = T2SUB + 1
                                                                              G1 08320
      T2PRED(T2SUB) = I
                                                                              G1 08330
                                                                              G1 08340
   49 CONTINUE
      NOW ALL CURRENT INPUTS HAVE ESSENTIAL 1'S
                                                                              G1 08350
0
C
      INPUTS STILL CONNECTED TO GCO ARE LISTED IN T2PRED IN REVERSE
                                                                              G1 08360
C
      ORDER
                                                                              G1 08370
C
                                                                              G1 08380
C
      UPDATE G(I)'S OF THOSE GATES STILL CONNECTED TO GATE GCO
                                                                              G1 08390
                                                                              G1 08400
      DO 29 II=1,F$UB1
                                                                              G1 08410
      I = F$1(II)
                                                                              G1 08420
      CHOICE = -GSMALL(GCO,I)
                                                                              G1 08430
      IF(CHDICE.LT.100 JGD TO 61
                                                                              31 08440
      CHDICE = 0
                                                                              G1 08450
      DO 30 JJJ=1.NR
                                                                              G1 08460
      JJ = TORDER(JJJ)
                                                                              G1 08470
      IF(INC$MX(JJ,GCO).EQ.O)GO TO 30
                                                                              G1 08480
      IF(P$(1,(JJ-1)*N2+I).NE.1)GO TO 30
                                                                              G1 08490
      IF(JJ.LE.N)GO TO 29
                                                                              G1 08500
      IF (CHOICE.FQ.O) CHOICE=JJ
                                                                              G1 08510
      IF (GSMALL(JJ, I).GE.1)GOTO29
                                                                              G1 08520
   30 CCNTINUE
                                                                              31 08530
   61 \text{ GSMALL(CHOICE,I)} = 1
                                                                              G1 08540
      USED(CHOICE) = 1
                                                                              G1 08550
   29 CONTINUE
                                                                              G1 08560
      DD 32 I=1, N2
                                                                              G1 08570
      IF(GS MALL(GCO, I).LT.1)GO TO 32
                                                                              G1 08580
      DO 33 J=1.T2SUB
                                                                              G1 08590
      IF(GSMALL(T2PRED(J),I).EQ.0)GSMALL(T2PRED(J),I)=-100
                                                                              G1 08600
   33 CONTINUE
                                                                              G1 08610
   32 CONTINUE
                                                                              G1 08620
      GOTO5
                                                                              G1 08630
      END
                                                                              G1 08640
      SUBROUTINE DUTPUT (MATRIX, ARRAY)
                                                                              G1 08650
      IMPLICIT INTEGER*4(A-T,V-Z,$), REAL(U)
                                                                              G1 08660
                                                                              G1 08670
C
      DEFINITIONS OF 'COMMON' VARIABLES CAN BE FOUND IN MAIN PROGRAM.
                                                                              G1 08680
                                                                              G1 08690
      COMMON NEPMAX
                                                                              G1 08700
      NCMMOD
                N
                                                Δ
                                                                 В
                                                                              G1 08710
               R
                                N<sub>2</sub>
                                                                 NR
                                                N1
                                                                              G1 08720
     2
               NM
                                KFLAG
                                                JFLAG
                                                                 COST
                                                                              G1 08730
     3
               LEVM
                                NRN2
                                                NM1
                                                                 NN<sub>2</sub>
                                                                              31 08740
      COMMON
                ISUCC(40,40)
                                LISUCC (40)
                                                IPRED(40,40)
                                                               , LIPRED(40)
                                                                              G1 08750
                                              •
                INC $MX(40,40),
                                SUC$MX(40,40), P$(2,1280)
                                                               , UNAME (40)
                                                                              G1 08760
               GLEVEL(40)
                              , LGLIST (40)
                                                HLIST(40,40) ,
                                                                 TIME
                                              9
                                                                              G1 08770
      COMMON
               T
                              , RTCONN(100)
                                              , S
                                                                              G1 08780
                                                               , RSCONN(100)
      NCMMOD
                                                               ,F$1(32)
               IFLAG
                              , POINTA
                                              , ESS 1S(40)
                                                                              G1 08790
           ,F$UB1
                              .INPTCV(32)
                                                               , POINTC
                                              ,LISTC(40)
                                                                              G1 08800
```

```
, POINTL
                                                  , ORIGIN(40)
                                                                   , IPATH(40)
                                                                                   G1 0881(
     2
            . LISTL(40)
                                                  , VF$UB1
                                                                   , GSMALL (40, 32) G1 0882(
     3
            . POINTR
                                ,VF$1(32)
      COMMON
                 POTAB(200, 42), PPOTAB(40)
                                                  , LPOTAB (40)
                                                                   · NRPLC(2)
                                                                                   G1 08830
                                                  , IDXOE (32)
                                , IDX 0 (32)
            , RPLC(2,40)
                                                                   , IDX1(32)
                                                                                   G1 08840
     1
                                ,SUMP(32)
     2
            , IDX1E(32)
                                                  ,SETT1(32)
                                                                   , NOT 1
                                                                                   G1 08851
            , SETS1(40)
                                                  , SETS(40)
                                                                   , NOS
                                , NOS1
                                                                                   G1 08860
     3
            , STS
                                ,SUMS 2( 32)
                                                  , SETS2(200)
                                                                   , NOS2
                                                                                   G1 08871
     4
     5
            .LIP
                                                                   , KEYB
                                                                                   G1 08886
                                ,NOOE
                                                  , KEYA
            , NO0
                                ,NO1
                                                  .NO1E
                                                                    , $GT
                                                                                   G1 0889(
     6
            , $LTH
                                , SPW
                                                                   , GI
     7
                                                  . $NOE
                                                                                   31 08900
                                                                  , LMTS2
                                                 , NOS 1SV
                                NOTISV
                                                                                   G1 08910
      NOMMES
      DIMENSION UX(5), UY(5), UG(40), UF(40), ARRAY(40), ARRAY2(2,1280) G1 08920
      DIMENSION MATRIX(40.40)
                                                                                   G1 08931
      DATA UX / X1 , X2 , X3 , X4 , X5 /
                                                                                   G1 08940
       DATA UY/ 111, Y21, Y31, Y41, Y51/
                                                                                   G1 08950
                    1 , ,
                           21,1
                                 31,1 41,1
                                                 51,1
      DATA UF /1
                                                       61,1
                                                               71,1
                                                                      8 *
                                                                                   G1 08961
                          10', ' 11', ' 12', '
                    91,1
                                               13','
                                                      14 , 1
                                                             15'.'
                                                                    16
                                                                                   G1 0897(
     1
                   17', ' 18', ' 19', ' 20', ' 21', '
                                                      221,1
                                                             231,1
                                                                    241
                                                                                   G1 08980
     2
                   25',' 26',' 27',' 28',' 29',' 30','
                                                             31', ' 32'
     3
                                                                                    G1 08990
                 1 331,1 341,1 351,1 361,1 371,1 381,1 391,1 401/
                                                                                   G1 09000
                                                                                    31 09010
      DATA GMAX/40/
                                                                                    G1 09020
C
                                                                                    G1 0903#
      KEYXC=ARRAY(1)
      IF(KEYXC.NE.O) GO TJ 50
                                                                                   G1 09041
                                                                                    G1 09050
      DJ 1 GI=1.N
                                                                                    G1 09061
        JNAME(GI)=UX(GI)
    1 CONTINUE
                                                                                    31 09070
                                                                                    G1 09080
      SO TO 100
                                                                                    G1 09090
   50 CONTINUE
      L=N/2
                                                                                    31 09100
                                                                                   G1 09111
      DO 4 GI=1.L
                                                                                   G1 09121
       UNAME(GI)=UX(GI)
                                                                                    G1 09130
      UNAME (GI+L) =UY (GI)
                                                                                    31 09140
    4 CONTINUE
  100 CENTINUE
                                                                                    31 09150
                                                                                    G1 0916
      DO 2 GI=N1,GMAX
                                                                                    G1 09171
       UNA 4E (GI) = UF (GI-N)
                                                                                    G1 09180
    2 CONTINUE
                                                                                    G1 09190
      RETURN
C
                                                                                    G1 09200
                                                                                    G1 09210
       ENTRY LINE(L)
                                                                                    G1 09221
      DD 6 LL=1.L
                                                                                    G1 09231
        PPINT 5
                                                                                    31 09241
    5 FORMAT (1H )
                                                                                    G1 09250
    6 CONTINUE
       RETURN
                                                                                    G1 09260
5
                                                                                    G1 09271
                                                                                    G1 09281
      ENTRY PAGE
                                                                                    G1 0929(
       PHINT 7
    7 FORMAT (1H1)
                                                                                    G1 09300
                                                                                    G1 09310
      RETURN
                                                                                    G1 09320
0
                                                                                    G1 0933
      ENTRY CKT (MATRIX, AFRAY)
       PRINT 10
                                                                                    51 09341
                                                                                    G1 0935
   10 FORMAT(1H .8X, GATE .. LEVEL 1,6X, FED BY1/)
                                                                                    31 0936
      DO 20 GJ=N1.NR
                                                                                    G1 0937
       5=0
                                                                                   G1 0938
        DO 15 GI=1,NR
                                                                                    G1 0939
         IF (MATRIX(GI,GJ).EQ.O) GC TO 15
                                                                                    G1 09401
         G = G + 1
                                                                                    31 0941
         UG(G)=UNAME(GI)
```

```
C
C
C
C
0
C
C
C
C
C
C
000
C
C
C
C
C
C
C
C
C
```

```
G1 09420
   CONTINUE
15
                                                                     G1 09430
    IF(G.EQ.O) GO TO 18
                                                                     G1 09440
   PRINT 17, UNAME(GJ), ARRAY (GJ), (UG(GG), GG=1, G)
   FORMAT(1HO, 9X,A3,5X,"/",I2,"/",5X,35( A3))
                                                                     G1 09450
17
   GO TO 20
                                                                     G1 09460
                                                                     G1 09470
   PRINT 19. UNAME(GJ), ARRAY(GJ)
18
   FORMAT (1HO, 9X, A3, 5X, 1/1, I2, 1/1)
                                                                     G1 09480
19
20 CONTINUE
                                                                     G1 09490
                                                                     G1 09500
   RETURN
                                                                     G1 09510
  ENTRY TRUTH (ARRAY2, J)
                                                                     G1 09520
   IF(J.EQ.2) GD TO 36
                                                                     G1 09530
                                                                     G1 09540
   PRINT 35
35 FORMAT(11X.
                 *TRUTH TABLE*/)
                                                                     G1 09550
  GO TO 38
                                                                     G1 09560
36 PRINT 37
                                                                     G1 09570
37 FORMAT(11X, "REQUIREMENT TABLE")
                                                                     G1 09580
38 CONTINUE
                                                                     G1 09590
                                                                     G1 09600
  DO 40 GI=1,NR
   ILO=(GI-1)*N2+1
                                                                     G1 09610
                                                                     G1 09620
   IHI=ILO+N2-1
   PRINT 41, UNAME(GI), (ARRAY2(J,I), I=ILO,IHI)
                                                                     G1 09630
                                                                     G1 09640
40 CONTINUE
41 FORMAT (1HO, 9X,A3,^{\circ} = ^{\circ}, 32(I1,1X))
                                                                     G1 09650
                                                                     G1 09660
  RETURN
                                                                     G1 09670
  END
  SUBROUTINE PROCII(JAYFLG, KEIFLG, EYEFLG, LFLAG)
                                                                     G1 09680
  G1 09700
  JAYFLG INDICATES THE VERSION OF CRDERING USED
                                                                     G1 09710
  KEIFLG=1 INDICATES NORMAL PROCII
                                                                     G1 09720
  KEIFLG = 2 INDICATES PROCIV (VERSION B)
                                                                     G1 09730
  KEIFLG = 3 INDICATES PROCIP
                                                                     G1 09740
  EYEFLG INDICATES THE GATE OF FOCUS FOR PROCIV
                                                                     G1 09750
  SET EYEFLG = 0 FOR OTHER PROCEDURES
                                                                     G1 09760
  LFLAG IS USED FOR PROCIV AND PROC3
                                                                     G1 09770
  LFLAG = 1 MEANS IGNORE GATES OF SUBNETWORK T
                                                                     G1 09780
  LFLAG = 0 MEANS CALCULATE CSPF'S FOR GATES OF SUBNETWORK T
                                                                     G1 09790
  SET LFLAG = 0 FOR PROCII AND PROCIP
                                                                     31 09800
  LFLAG = -1 MEANS USE SPECIAL ORDERINGS PREFERING HIGH LEVEL GATES G1 09810
  GESTED FOR PROC3 - DEFINITELY NOT FOR PROCIV)
                                                                     G1 09820
                                                                     G1 09830
  IMPLICIT INTEGER*4(A-T.V-Z.$), REAL(U)
                                                                     G1 09840
                                                                     G1 09850
  DEFINITIONS OF "COMMON" VARIABLES CAN BE FOUND IN MAIN PROGRAM.
                                                                     G1 09860
                                                                     G1 09870
  VARIABLE DEFINITIONS:
                                                                     G1 09880
       C: COMPONENT POSITION OF A 'O' IN GCO'S CSPF.
                                                                     31 09890
  CAFTER: NO. OF CONNECTIONS AFTER TRANSFORMATION.
                                                                     G1 09900
  CBEFOR: NO. OF CONNECTIONS BEFORE TRANSFORMATION.
                                                                     G1 09910
     CCO: AN EFFECTIVELY CONNECTIBLE FUNCTION CHOSEN TO BE ADDED TO G1 09920
          GCO AS A NEW INPUT.
                                                                     G1 09930
  CHOICE: AN INPUT CHOSEN TO COVER A "O" COMPONENT.
                                                                     G1 09940
  COVERD: COVERD(GI) TELLS NO. OF O'S IN CSPF VECTOR OF GCO COVERED G1 09950
          BY GI.
                                                                     G1 09960
     CVD: NO. OF ZEROS IN CSPF VECTOR OF GCO COVERED BY AN INPUT I. G1 09970
  EFFCON: A SELECTED EFFECTIVELY CONNECTIBLE FUNCTION.
                                                                     G1 09980
  ESTEMP: TEMPORARY STORAGE USED TO CALCULATE "ESSIS".
                                                                     31 09990
   F$UBO: NUMBER OF NECESSARY ZEROS LISTED IN F$O.
                                                                     G1 10000
```

```
F&O: LISTS (CONSECUTIVELY) POSITIONS OF NECESSARY ZEROS IN A
                                                                            G1 10010
              CONNECTIBLE FN.
                                                                            G1 10020
      GAFTER: NO. OF GATES IN NETWORK AFTER TRANSFORMATION.
                                                                            G1 10030
      GREEDE: NO. DE GATES IN NETWORK BEFORE TRANSFORMATION.
                                                                            G1 10040
         SCC: THE GATE (OR EX. VAR.) IN GORDER POINTED TO BY GCOUNT.
                                                                            G1 10050
      GCOUNT: A POINTER TO CURRENT POSITION IN GORDER.
                                                                            G1 10060
      GORDER: CONTAINS AN ORDERED LIST OF GATES AND EXTERNAL VARIABLES. G1 10070
      IMPROV: PARAMETER RETURNED BY MINI2 (UNUSED BY PROCII).
                                                                            G1 10080
C
      INSERT: POSITION IN LISTO TO INSERT NEWCOMERS.
                                                                            G1 10090
        IPOS: POINTS TO GATE EYEFLG IN GORDER (FOR PROCIV).
                                                                            31 10100
      IPOSP1: EQUALS IPOS+1.
                                                                            51 10110
      NEWCST: COST OF NEW NETWORK.
                                                                            G1 10120
C
      NMINLV: NO. OF GATES AND EX. VAR. IN A CERTAIN LEVEL OF NETWORK.
                                                                            G1 10130
      OLDCST: COST OF ORIGINAL NETWORK.
                                                                            G1 10140
      ORDERP: ORDERING OF GCO'S INPUTS ONLY - IN SAME ORDER AS RORDER.
                                                                            G1 10150
           P: A CONNECTIBLE FN. TO REPLACE INPUT TH TO GCO.
                                                                            31 10160
       PICK1: FIRST PREFERENCE COVER.
                                                                            G1 10170
       PICK2: SECOND PREFERENCE COVER.
                                                                            G1 10180
C
       PICK3: THIRD PREFERENCE COVER.
                                                                            G1 10190
       PICK4: FOURTH PREFERENCE COVER.
                                                                            G1 10200
       POINT: POINTS TO LAST ENTRY IN ORDERP.
                                                                            G1 10210
      POINTT: NO. OF ELEMENTS IN THESS.
                                                                            G1 10220
      QINC $M: SAVES A COPY OF ORIGINAL INC $MX.
                                                                            G1 10230
Ç
      RORDER: AN ORDERING OF GATES AND EX. VARS. DERIVED FROM GORDER
                                                                            G1 10240
              ISTRUCTURE OF ORDERING DIFFERS ACCORDING TO KEIFLG AND
C
                                                                            G1 10250
              LFLAGI.
                                                                            G1 10260
      SAVING: COST IMPROVEMENT OF NEW NETWORK OVER ORIGINAL (CAN BE
                                                                            G1 10270
              POSITIVE OR NEGATIVE).
                                                                            G1 10280
        SUBL: POINTER TO ELEMENTS OF LISTL.
                                                                            G1 10290
          TH: AN INPUT TO GCC SELECTED FOR POSSIBLE REPLACEMENT.
                                                                            G1 10300
       THESS: LIST OF ESSENTIAL 1'S IN A PARTICULAR INPUT (TH) TO GCO.
                                                                            G1 10310
      THROWN: EQUAL TO NO. OF ENTRIES THROWN OUT OF LISTC WHEN A
                                                                            G1 10320
              CERTAIN NEW ENTRY IS CONSIDERED FOR INSERTION.
                                                                            G1 10330
                                                                            G1 10340
      CDUNT, I, II, J, JJ, K, KK, LAST, TEMP, TEND, TEST, X, XY, Y, Z, ZFLAG, ZONE, ZZ
\Gamma
                                                                            G1 10350
      ARE USED AS JUST TEMPCRARY VARIABLES.
                                                                            G1 10360
                                                                            G1 10370
      HOW TO INCREASE CAPACITY OF SUBROUTINE.
                                                                            G1 10380
      DIMENSION: COVERD(X), RORDER(X),
                                                                            G1 10390
                  DINCSM(X,X), ORDERP(X) - X EQUAL TO MAX NUMBER OF GATES G1 10400
                                           PLUS EXTERNAL VARIABLES.
                                                                            G1 104I0
                 GORDER(Y) - Y EQUAL TO X+1
                                                                            G1 10420
                  ESTEMP(Z), F$0(Z), THESS(Z) - Z EQUAL TO 2**(MAX ALLOWED G1 10430
                                               NO. OF EX. VAR.)
                                                                            31 10440
                                                                            G1 10450
      **** NOTE PROGRAM SECTIONS DJ NOT CORRESPOND EXACTLY IN NAME OR
                                                                            G1 10460
           NUMBER TO FLOWCHART BLOCKS IN USER'S MANUAL. *****
                                                                            G1 10470
                                                                            G1 10480
                                                                            G1 10490
                                                                            G1 10500
      CUMMON NEPMAX
      COMMON
                             , M
                                                              , B
                                                                            G1 10510
               V
                                               Α
                                                             . NR
               2
                             , N2
                                                                            31 10520
                                             , N1
     1
           9
                                                                            G1 10530
     2
               MM
                             , KFLAG
                                             . JFLAG
                                                             , COST
           9
                             · NRN2
                                                             , NN2
               LEVM
                                              . NM1
                                                                            G1 10540
      COMMON
               ISUCC(40,40) , LISUCC(40)
                                           , IPRED(40,40) , LIPRED(40)
                                                                            G1 10550
               INC$MX(40,40), SUC$MX(40,40), P$(2,1280)
                                                             , UNAME(40)
                                                                            G1 10563
     9
           9
                                                                            G1 10570
               SLEVEL (40)
                               LGLIST(40)
                                             . HLIST(40,40) .
                                                               TIME
                                                             , RSCCNN(100) G1 10580
      COMMON
                             , RTCCNN(100)
               Ţ
                                             , 5
      NOMMOD
              IFLAG
                             , POINTA
                                                                            G1 10590
                                             , ESS1S(40)
                                                             ,F$1(32)
```

,INPTCV(32)

, POINTL

, LISTC (40)

ORIGIN(40)

\*PUINTC

, IPATH(40)

G1 10600

G1 10610

,F&UBI

·LISTL(40)

```
,VF$1(32)
            . POINTR
                                               .VF $UB1
                                                               GSMALL(40,32)G1 10620
     3
      COMMON
                POTAB(200,42), PPOTAB(40)
                                               . LPOTAB(40)
                                                               .NRPLC(2)
                                                                              G1 10630
                              , IDX0(32)
                                               , IDXOE (32)
                                                               , IDX1(32)
                                                                              G1 10640
           , RPLC(2,40)
                                                                              G1 10650
            . IDX1E(32)
                              ,SUMP(32)
                                               ,SETT1(32)
                                                               ,NOT1
     2
            . SETS1(40)
                              . NOS1
                                               .SETS(40)
                                                               . NOS
                                                                              G1 10660
     3
                                               , SETS2(200)
            , STS
                              , SUMS 2(32)
                                                               ,NOS2
                                                                              G1 10670
                                                                              G1 10680
                                               , KEYA
                                                               , KEYB
            .LIP
                              , NOOE
     5
                                                                              G1 10690
                              , NO1
                                               ,NOIE
                                                               , $GT
            .NO0
     6
            . SLTH
                              . SPW
                                               . $NOE
                                                               .GI
                                                                              G1 10700
      COMMON
                                                              .LMTS2
                                                                              G1 10710
                              NOT1 SV
                                              .NOS1SV
                                               ,ESTEMP(32)
                                                               ,F$0(32)
                               COVERD(40)
                                                                              G1 10720
      DIMENSION
                                               ,QINC$M(40,40) ,THESS(32)
            , GORDER (41)
                              .RORDER(40)
                                                                              G1 10730
           , ORDERP(40)
                                                                              G1 10740
     2
                                                                              G1 10750
      IF(KEIFLG.NE.2)GOTO 185
      IF(LISUCC(EYEFLG).EQ.O)RETURN
                                                                              G1 10760
                                                                              G1 10770
C
                                                                              31 10780
C
      SECTION 1, START
                                                                              G1 10790
  185 CALL SUBNET
                                                                              G1 10800
      CALL PVALUE
                                                                              G1 10810
                                                                              G1 10820
      CALL UNNECE
                                                                              G1 10830
      KFLAG = KEIFLG
      JFLAG = JAYFLG
                                                                              G1 10840
      IFLAG = EYEFLG
                                                                              G1 10850
C
      CALCULATE NUMBER OF GATES ACTUALLY IN NETWORK
                                                                              G1 10860
      GBEFOR = M
                                                                              G1 10870
      CBEFOR = 0
                                                                              G1 10880
      DO 167 I=1.NR
                                                                              G1 10890
      CBEFOR = CBEFOR + LISUCC(I)
                                                                              31 10900
      IF(I.LE.NM) GO TO 167
                                                                              G1 10910
      IF(LISUCC(I).GT.O) GBEFOR = GBEFOR + 1
                                                                              G1 10920
  167 CONTINUE
                                                                              31 10930
      IF (KFLAG.NE.2) GOTO186
                                                                              G1 10940
                                                                              G1 10950
      DO 187 I=1,NR
      IPATH(I)=0
                                                                              GI 10960
      IF(SUC$MX(IFLAG, I).GT.O)IPATH(I)=1
                                                                              G1 10970
  187 CONTINUE
                                                                              G1 10980
      IPATH(IFLAG)=1
                                                                              31 10990
      MAKE A COPY OF INCSMX IN "QINCSM"
                                                                              G1 11000
  186 DO 132 I=1,NR
                                                                              G1 11010
C
      INITIALIZE GSMALL WITH DCN'T CARES -
                                                                              G1 11020
C
          GSMALL RECORDS FOR EACH GATE THE SUMMATION OF COMPONENTS
                                                                              G1 11030
C
          OF ALL FUNCTIONS (G(I,J)) REQUIRED BY LINES FROM GATE I
                                                                              G1 11040
C
          TO EVERY SUCCESSOR J.
                                                                              G1 11050
      DO 50 J=1, N2
                                                                              G1 11060
   50 GSMALL (I.J) = 0
                                                                              G1 11070
      D3 132 J=1,NR
                                                                              G1 11080
  132 QINC$M(I,J) = INC$MX(I,J)
                                                                              G1 11090
C
                                                                              31 11100
      SECTION 2, ORDER GATES AND EXTERNAL VARIABLES
                                                                              G1 11110
C
                                                                              G1 11120
    2 ZONE = 0
                                                                              G1 11130
      COUNT=1
                                                                              G1 11140
  189 DJ 101 I=1, LEVM
                                                                              G1 11150
      NMINLV=LGLIST(I)
                                                                              G1 11160
      IF(NMINLV.EQ.O)GOTO101
                                                                              G1 11170
      DO 102 J=1, NMINLV
                                                                              G1 11180
      K = HLIST(J,I)
                                                                              G1 11190
      IF(KFLAG.EQ.2.AND.IPATH(K).EQ.ZONE) GO TO 102
                                                                              G1 11200
      GORDER (COUNT)=K
                                                                              G1 11210
      IF(K .EQ.IFLAG) GO TO 133
                                                                              G1 11220
```

```
G1 11230
       COUNT = COUNT +1
  102 CONTINUE
                                                                                 G1 11240
  101 CONTINUE
                                                                                 G1 11250
                                                                                 G1 11260
       IF(CJUNT.EQ.NR+1)GOTO160
  133 GORDER (COUNT) = IFLAG
                                                                                 G1 11270
       IPOS = COUNT
                                                                                 G1 11280
       COUNT = COUNT + 1
                                                                                 G1 11290
       ZONE = 1
                                                                                 G1 11300
       GO TO 189
                                                                                 G1 11310
       FORM ROPDER
                                                                                 G1 11320
  160 IF (KFLAG.EQ. 2) GO TO 161
                                                                                 G1 11330
       IF(LFLAG.EQ.-1) GO TO 214
                                                                                 G1 11340
       00 162 I = 1, NR
                                                                                 G1 11350
  162 RORDER(I) = GORDER(I)
                                                                                 G1 11360
       GO TO 3
                                                                                 G1 11370
  214 D0 215 I = 1.NR
                                                                                 G1 11380
  215 RORDER(I) = GORDER(NR+1-I)
                                                                                 G1 11390
                                                                                 G1 11400
      GOTO 3
  161 \text{ IPOSP1} = \text{IPOS} + 1
                                                                                 G1 11410
       DO 163 I = IPOSP1,NR
                                                                                 G1 11420
  163 RORDER(I - IPOS) = GORDER(I)
                                                                                 31 11430
       J = NR - IPOS
                                                                                 G1 11440
       DO 169 I = 1, IPOS
                                                                                 G1 11450
  169 RORDER(J+I) = GORDER(I)
                                                                                 G1 11460
                                                                                 G1 11470
C
C
      SECTION 3, CALCULATE F(I)
                                                                                 G1 11480
Ċ
                                                                                 G1 11490
    3 LAST=N2*NR
                                                                                 G1 11500
      DO 104 I=1, LAST
                                                                                 G1 11510
                                                                                 G1 1152(
  104 P$(2,I) = P$(1,I)
       INITIALIZE GSMALL FOR OUTPUT GATES
                                                                                 G1 1153(
      DO 53 I=N1, NM
                                                                                 31 11540
      X = (I - 1) * N2
                                                                                 G1 11550
                                                                                 G1 11560
      DO 53 J=1, N2
      Y = P\$\{2,X+J\}
                                                                                 G1 11570
      IF(Y.EQ.O) GSMALL(I,J) = -100
                                                                                 G1 11580
       IF(Y.EQ.1) GSMALL(I,J) = 1
                                                                                 G1 11590
                                                                                 G1 11600
       IF(Y.EQ.-1) GSMALL(I,J) = 0
   53 CONTINUE
                                                                                 G1 1161(
C
                                                                                 G1 11624
C
       SECTION 4, INITIALIZE COUNTER TO LOOP ONCE FOR EACH GATE
                                                                                 G1 11631
C
                                                                                 G1 11640
    4 GC JUNT = 0
                                                                                 G1 1165
                                                                                 G1 1166
C
00
                                                                                 G1 1167
      SECTION 5, INCREMENT COUNT
                                                                                 G1 1168
5
      GCOUNT=GCOUNT+1
                                                                                 G1 1169
Ç
                                                                                 G1 1170
C
      SECTION 6. ARE ALL GATES EXHAUSTED?
                                                                                 G1 1171
                                                                                 G1 1172
                                                                                 G1 1173
    6 IF(LFLAG.EQ.1.AND.GCOUNT.EQ.IPOS)GOTO7
                                                                                 G1 1174
       IF (GCQUNT.GT.NR)GOT 97
      GC D=G DRDER (GCOUNT)
                                                                                 G1 1175
                                                                                 G1 1176
      IF(GCC.LE.N) GC TO 33
C
                                                                                 31 1177
C
      SECTION 11, CALCULATE G(GCO)
                                                                                 G1 1178
                                                                                 31 1179
11
      X = N2*(GCO-1)
                                                                                 G1 1180
      00 190 I=1,N2
                                                                                 G1 1181
      Y = GSMALL(GCO, I)
                                                                                 G1 1182
                                                                                 G1 1183
      IF(Y.EQ.0) P$(2,X+I) = -1
```

```
IF(Y.GT.O) P$(2,X+I) = 1
                                                                          G1 11840
    IF(Y.LT.0) P$(2,X+I) = 0
                                                                          G1 11850
                                                                          G1 11860
190 CONTINUE
                                                                          G1 11870
    SECTION 33. IS GCO AN ISOLATED GATE OR AN EXTERNAL VARIABLE ?
                                                                          G1 11880
      IF ISOLATED: REMOVE ANY INPUTS, UPDATE ARRAYS, RETURN TO SEC. 5 G1 11890
      IF EXTERNAL VARIABLE: RETURN TO SECTION 5
                                                                          G1 11900
                                                                          G1 11910
33 IF(GCO.LE.N)GO TO 5
                                                                          G1 11920
    DO 54 I=1, N2
                                                                          G1 11930
                                                                          G1 11940
    IF(GSMALL(GCO, I).GE.1)GOTO12
                                                                          G1 11950
 54 CONTINUE
    IF HERE. THEN GATE IS NONESSENTIAL - REMOVE INPUTS AND OUTPUTS
                                                                          G1 11960
    X = LISUCC(GCO)
                                                                          G1 11970
                                                                          G1 11980
    IF(X.EQ.0)GO TO 201
    DO 202 I=1,X
                                                                          G1 11990
    Y = ISUCC(1,GCO)
                                                                          G1 12000
   CALL RNONES (Y, GCO, O)
                                                                          G1 12010
202 CONTINUE
                                                                          G1 12020
201 X=LIPRED(GCO)
                                                                          G1 12030
                                                                          G1 12040
    IF(X.EQ.O)GO TO 5
                                                                          G1 12050
   D3 55 I=1.X
                                                                          G1 12060
   Y = IPRED(1,GCO)
   CALL PNONES (GCD, Y, 0)
                                                                          G1 12070
                                                                          G1 12080
55 CONTINUE
   GO TO 5
                                                                          G1 12090
                                                                          G1 12100
   SECTION 12, CALCULATE F(GCO)
                                                                          G1 12110
                                                                          G1 12120
12 Z=N2*(GCO-1)
                                                                          G1 12130
   F$UB0=1
                                                                          G1 12140
    F$UB1=1
                                                                          G1 12150
   DO 51 I=1,N2
                                                                          G1 12160
                                                                          G1 12170
    X=GSMALL(GCO, I)
    IF(X.EQ. 0)GO TO 51
                                                                          G1 12180
    IF(X.LT. 0)GO TO 52
                                                                          G1 12190
   F$0(F$U80)=I
                                                                          G1 12200
   F$UB0=F$UB0+1
                                                                          G1 12210
   GO TO 51
                                                                          G1 12220
52 F$1(F$UB1)=I
                                                                          G1 12230
   VF$1( F$UB1) = I
                                                                          G1 12240
                                                                          G1 12250
   F$UB1=F$UB1+1
51 CONTINUE
                                                                          G1 12260
   F$UB0=F$UB0-1
                                                                          G1 12270
   F$U81=F$U81-1
                                                                          G1 12280
   VF$UB1 = F$UB1
                                                                          G1 12290
                                                                          G1 12300
   SECTION 13, THIS SECTION CONTAINS SUBSECTIONS 15 THROUGH 32 + 34
                                                                          G1 12310
                                                                          31 12320
      THIS SECTION WILL ADD CONNECTIBLE FUNCTIONS, REMOVE ANY
                                                                          G1 12330
     UNNECESSARY INPUTS, UPDATE G(I,GCO) FOR ALL I, WHERE I FEEDS GCOG1 12340
                                                                          G1 12350
                                                                          G1 12360
   SECTION 34, CALCULATE HOW MANY INPUTS COVERING EACH COMPONENT
                                                                          G1 12370
                                                                          G1 12380
   DO 88 I=1,F$UB1
                                                                          G1 12390
88 INPTCV(F$1(I))=0
                                                                          G1 12400
   X = LIPRED(GCO)
                                                                          G1 12410
   DO 57 I=1, X
                                                                          G1 12420
   XY = IPRED(I,GCO)
                                                                          G1 12430
    Y = N2*(XY-1)
                                                                          G1 12440
```

CC

C

C

C

C

C

00000

C

C

0

```
D3 89 J=1, F$UB1
                                                                               G1 12450
      Z = F$1(J)
                                                                               G1 12460
      IF(P$(2,Y+Z).EO.1)INPTCV(Z)=INPTCV(Z)+1
                                                                               G1 12470
   89 CONTINUE
                                                                               G1 12480
   57 CONTINUE
                                                                               G1 12490
      INITIALIZE ORIGIN(40)
                                                                               G1 12500
      DO 150 I=1,NR
                                                                               G1 12510
      IF(INC$MX(I,GCC).EQ.O)GO TO 151
                                                                               G1 12520
      ORIGIN(I) = 1
                                                                               G1 12530
      GO T 1 150
                                                                               G1 12540
  151 ORIGIN(I) = 0
                                                                               G1 12550
  150 CONTINUE
                                                                               G1 12560
C
                                                                               G1 12570
                                                                               G1 12580
C
      SECTION 15, LIST NUMBER OF ESSENTIAL CNES IN EACH INPUT TO GCO
                                                                               G1 12590
                                                                               G1 12600
   15 DO 59 I=1.F$UB1
                                                                               G1 12610
   59 ESTEMP(I)=0
      X=LIPPED(GCO)
                                                                               31 12620
      D7 172 J=1.X
                                                                               G1 12630
      XY=IPRED(J.GCO)
                                                                               G1 12640
                                                                               G1 12650
      Y=N2*(XY-1)
      PO 56 I=1,F$UB1
                                                                               G1 12660
      Z = F $1 (I)
                                                                               G1 12670
      TEMP=P s(2,Y+Z)
                                                                               G1 12680
      IF (TEMP.LE.O)GCT056
                                                                               G1 12690
      TEST=ESTEMP(I)
                                                                               G1 12700
                                                                               G1 12710
      IF(TEST.LT.0)GOT056
                                                                               G1 12720
      IF (TEST.GT.O)GOTO58
                                                                               G1 12730
      ESTEMP(I)=XY
      GO TO 56
                                                                               G1 12740
   58 \text{ ESTEMP}(I) = -1
                                                                               G1 12750
   56 CONTINUE
                                                                               G1 12760
  172 CONTINUE
                                                                               G1 12770
                                                                               G1 12780
      D3 60 I=1.X
   60 ESSIS(IPRED(I,GCD))=0
                                                                               G1 12790
                                                                               G1 12800
      00 61 I=1,F$UB1
                                                                               G1 12810
      Y=ESTEMP(I)
      IF(Y.LE.0)GO TO 61
                                                                               G1 12820
      FSSIS(Y) = ESSIS(Y) + 1
                                                                               G1 12830
   61 CONTINUE
                                                                               G1 12840
C
                                                                               G1 12850
                                                                               G1 12863
      SECTION 16, ELIMINATE NON-ESSENTIAL INPUTS AND ORDER OTHERS (IN
                                                                               G1 12870
C
        "LISTL") BY DECREASING NUMBER OF ESSENTIAL CNES
                                                                               G1 1288(
C
   16 CALL ELANDO(GCO)
                                                                               G1 12890
C
                                                                               G1 12900
      SECTION 17, LIST ALL CONNECTIBLE INPUTS TO GCO (IN LISTC) IN ORDERG1 1291(
_
                                                                              G1 12920
         OF DECREASING NUMBER OF ZEROS IN G(GCO) COVERED. DO NOT LIST
C
          ANY SMALLER THAN OTHERS.
                                                                               G1 12930
                                                                               G1 12940
5
      SEARCH FOR CONNECTIBLE INPUTS THAT COVER AT LEAST ONE O
                                                                               G1 12950
   17 POINTC=0
                                                                               31 12960
      IF(KFLAG.EQ.3)GOTO32
                                                                               G1 1297(
                                                                               G1 12980
      DO 71 I=1, NR
                                                                               G1 12990
C
      SATE I MUST NOT BE A SUCCESSOR OF GATE GCO
      IF(SUC$MX(GCO,I).GT.0)GGTO71
                                                                               G1 1300(
      GATE I MUST NOT FEED (NOW OR PREVIOUSLY) GATE GCO
                                                                              G1 1301(
C.
                                                                              G1 13020
      IF (QINC $M(I,GCO).GT.O)GOTC71
                                                                              G1 1303(
      Y=N2*(I-1)
C
                                                                              G1 1304(
      GATE I MUST HAVE ALL THE NECESSARY O'S
                                                                              G1 1305(
      DO 72 J=1,F$UBO
```

```
G1 13060
    IF(P$(2,F$0(J)+Y).NE.0)GOTO71
                                                                         G1 13070
 72 CONTINUE
    GATE I MUST COVER AT LEAST ONE O IN GCO
                                                                         G1 13080
                                                                         G1 13090
    DO 73 J=1,F$UB1
    IF(P$(2,F$1(J)+Y).EQ.1)GOTC74
                                                                         G1 13100
                                                                         G1 13110
 73 CONTINUE
                                                                         G1 13120
    G0T071
 74 CONTINUE
                                                                         G1 13130
    GATE I MUST NOT BE ISOLATED IF NOT AN OUTPUT GATE
                                                                        G1 13140
                                                                        G1 13150
    IF(LISUCC(I).EQ.O.AND.(I.GT.NM
                                              11G0T071
                                                                         G1 13160
    IF PROCIV IN EFFECT, CONNECTIBLE CANNOT BE ON
    PATH FROM GATE IFLAG TO OUTPUT
                                                                         G1 13170
    IF (KFL AG. EQ. 2. AND. IPATH(I) . EQ. 1)GOTO71
                                                                        G1 13180
    TRY TO PLACE GATE NUMBER INTO CORRECT POSITION IN LISTC
                                                                        G1 13190
                                                                         G1 13200
    ZFLAG = 1
                                                                         G1 13210
    ZZ = (I-1)*N2
                                                                         G1 13220
    IF(PDINTC.NE.O)GOTO75
    CASE WHEN LISTC IS EMPTY
                                                                         G1 13230
                                                                         G1 13240
    ZFLAG = 0
    POINTC = 1
                                                                         G1 13250
                                                                         GI 13260
    LISTC(1) = I
    CALCULATE NUMBER OF ZEROS IN G(GCO) COVERED BY I
                                                                         G1 13270
                                                                         G1 13280
 75 \text{ CVD} = 0
    00 77 J=1,F$UB1
                                                                         G1 13290
    IF(P$(2,F$1(J)+ZZ).NE.1)GOTO77
                                                                         G1 13300
                                                                         G1 13310
    CVD= CVD+1
 77 CONTINUE
                                                                         G1 13320
    COVERD(I) = CVD
                                                                         G1 13330
                                                                         G1 13340
    IF(ZFLAG.EQ.O)GOTO71
                                                                         G1 13350
    INSERT = 0
    THROWN = 0
                                                                         G1 13360
    DO 76 JJ=1, POINTC
                                                                         31 13370
    J = JJ - THROWN
                                                                         G1 13380
    TEST IF ENTRY IS .GT. OR .LT. NEW INPUT
                                                                         G1 13390
    Y= COVERD(LISTC(J))
                                                                         G1 13400
                                                                         G1 13410
    Z = (LISTC(J)-1)*N2
    IF ( INS ERT . NE . 0 ) GOTO 78
                                                                         G1 13420
    IF(Y.LE.CVD)INSERT = J
                                                                         31 13430
 78 CONTINUE
                                                                         G1 13440
    IF(CVD.LE.Y)GO TO 79
                                                                         G1 13450
    TEST HERE IF VECTOR CORRES. TO J IS SMALLER THAN VEC. CORRES. TO IG1 13460
                                                                         G1 13470
    DO 81 K=1,F$U81
                                                                         G1 13480
    IF(P$(2,Z+F$1(K)).GT.P$(2,ZZ+F$1(K)))GO TO 76
 81 CONTINUE
                                                                         G1 13490
    IF(INSERT.NE.J)GO TO 82
                                                                         G1 13500
    IF HERE, WE CAN SIMPLY EXCHANGE I WITH J
                                                                         G1 13510
    INSERT = -1
                                                                         G1 13520
    LISTC(J) = I
                                                                         G1 13530
   GO TO 76
                                                                         G1 13540
 82 TEND = POINTC-THROWN-1
                                                                         G1 13550
    IF (TEND.EQ.0)GOTO200
                                                                         G1 13560
    DO 83 K=J, TEND
                                                                         G1 13570
83 LISTC(K)= LISTC(K+1)
                                                                         G1 13580
200 THROWN = THROWN + 1
                                                                         G1 13590
    GO TO 76
                                                                         G1 13600
    TEST HERE IF VEC. CORRES. TO J IS .EQ. TO VEC. CORRES. TO I
                                                                         G1 13610
 79 IF(CVD.LT.Y)GO TO 80
                                                                         G1 13620
    DO 84 K=1,F$UB1
                                                                         G1 13630
    IF(P$(2,Z+F$1(K)).NE.P$(2,ZZ+F$1(K)))GO TO 76
                                                                         G1 13640
 84 CONTINUE
                                                                         31 13650
    IF HERE, VECTORS FOR I AND J ARE IDENTICAL, CNLY INSERT IF I IS
                                                                         G1 13660
```

C

C

C

C

C

C

C

C

C

C

```
C
         GATE AND J IS EX. VAR.
                                                                             G1 13670
      IF(I.LE.N.OR.LISTC(J).GE.N1)GO TO 71
                                                                             G1 13680
      LISTC(J)=I
                                                                             G1 13690
      GO TO 71
                                                                             G1 13700
      TEST HERE IF VECTOR CORRES. TO J LARGER THAN VEC. CORRES. TO I
                                                                             G1 13710
   80 DO 35 K=1,F$UB1
                                                                             G1 13720
      IF(P$(2,Z+F$1(K)).LT.P$(2,ZZ+F$1(K)))GO TO 76
                                                                             G1 13730
   85 CONTINUE
                                                                             G1 13740
                                                                             31 13750
      IF HERE, WE CAN THROW OUT I
      GO TO 71
                                                                             G1 13760
   76 CONTINUE
                                                                             G1 13770
      POINTC = POINTC - THROWN
                                                                             G1 13780
      NOW INSERT NEW INPUT INTO LISTC
C
                                                                             G1 13790
      IF (INSERT.LT.0)GO TO 71
                                                                             31 13800
                                                                             G1 13810
      IF (INSERT. GT. 0) GO TO 86
C
      IF HERE, ADD NEW INPUT TO END OF LISTC
                                                                             G1 13820
      POINTC = POINTC + 1
                                                                             G1 13830
      LISTC(POINTC)=I
                                                                             G1 13840
      GO TO 71
                                                                             G1 13850
C
      SHIFT AND INSERT
                                                                             G1 13860
   86 POINTC = POINTC + 1
                                                                             G1 13870
      DO 87 J=INSERT, POINTC
                                                                             G1 13880
      JJ = PCINTC-(J-INSERT)
                                                                             31 13890
                                                                             G1 13900
   87 LISTC(JJ+1)=LISTC(JJ)
      LISTC(INSERT) = I
                                                                             G1 13910
                                                                             31 13920
   71 CONTINUE
      IF(POINTC.EQ.O)GOTO32
                                                                             G1 13930
C
                                                                             G1 13940
Ç
      SECTION 18, SELECT TOP ELEMENT, (CALL IT "TH") FROM LISTL
                                                                             G1 13950
                                                                             G1 13960
   18 IF(POINTC.EQ.O)GOTO 32
                                                                             G1 13970
      TH = LISTL(1)
                                                                             G1 13980
      SUBL = 1
                                                                             G1 13990
                                                                             31 14000
      SECTION 19, SEARCH LISTC FOR A REPLACEMENT FOR TH
                                                                             G1 14010
                                                                             G1 14020
                                                                             G1 14030
      LIST ESSENTIAL CNES OF TH
   19 POINTT=0
                                                                             G1 14040
      Y = (TH-1)*N2
                                                                             G1 14050
      00 90 I=1.F$UB1
                                                                             G1 14060
                                                                             G1 14070
      X = F$1(I)
      IF(INPTCV(X).NE.1)GOTO90
                                                                             G1 14080
      IF(P$(2,Y+X).NE.1)GOT090
                                                                             G1 14090
                                                                             G1 14100
      POINTT = POINTT+1
      THESS(POINTY) = X
                                                                             G1 14110
   90 CONTINUE
                                                                             G1 14120
C
      NOW SEARCH LISTO
                                                                             G1 14130
      DO 91 I=1.POINTO
                                                                             31 14140
C
      IF CURRENT ELT. OF LISTC BEING CHECKED HAS FEWER COVERING 1'S THANG1 14150
         TH HAS ESSENTIAL 1°S, THEN NO REPLACEMENT, P. CAN BE FOUND IN G1 14160
C
         REMAINDER OF LISTC- SO SKIP TO SECTION 21
                                                                             G1 14170
      IF(COVERD(LISTC(I)).LT.POINTT)GO TO 21
                                                                             G1 14180
      Y = N2*(LISTC(I)-1)
                                                                             G1 14190
      DO 92 J=1, POINTT
                                                                             G1 14200
      X = THESS(J)
                                                                             G1 14210
      IF(P%(2,Y+X).NE.1)GOTO91
                                                                             G1 14220
                                                                             G1 14230
   92 CONTINUE
      IF HEPE, GATE (OR EX. VAR.) LISTC(I) IS A REPLACEMENT, P, FOR TH
                                                                            G1 14240
                                                                             31 14250
      P = LISTC(I)
                                                                             G1 14260
      GO TO 23
   91 CHITINUE
                                                                             31 14270
```

```
GO TO 21
                                                                              G1 14280
                                                                              G1 14290
C
      SECTION 23, A REPLACEMENT (CONNECTIBLE FN.), P, HAS BEEN FOUND
                                                                              G1 14300
C
        TO REPLACE TH (CURPENTLY AN INPUT TO G(GCO) - SO DISCONNECT TH
C
                                                                              G1 14310
                                                                              G1 14320
C
   23 CALL RNONES (GCD.TH.
                                  21
                                                                              G1 14330
                                                                              G1 14340
C
      SECTION 30, CONNECT REPLACEMENT GATE, P. TO GCD
                                                                              G1 14350
C
   30 CALL CNCCC(GCO,P)
                                                                              G1 14370
      IF(PCINTL.EQ.O) GO TO 24
                                                                              G1 14375
      UPDATE ESSIS AND INPTCV
                                                                             G1 14380
      DO 118 I=1.F$UB1
                                                                              G1 14390
      C = F$1(I)
                                                                              61 14400
      X = (TH-1)*N2
                                                                              G1 14410
      Y = (P - 1) * N2
                                                                              G1 14420
      IF(P$(2,C+X).EQ.1)GD TO 119
                                                                              G1 14430
      IF(P$(2,C+Y).EQ.1)GO TO 120
                                                                              G1 14440
      IF HERE, NO CHANGE NEEDED FOR COMPONENT C
                                                                              G1 14450
C
      GO TO 118
                                                                             G1 14460
  120 \text{ INPTCV(C)} = \text{INPTCV(C)}+1
                                                                              G1 14470
      IF(INPTCV(C).GT.2)GO TO 118
                                                                             G1 14480
                                                                             G1 14490
      DO 121 J=1, POINTL
      Z = (LISTL(J)-1)*N2
                                                                             G1 14500
      IF (P$(2,Z+C).NE.1)GO TO 121
                                                                             G1 14510
      ESSIS(LISTL(J)) = ESSIS(LISTL(J)) - 1
                                                                             61 14520
                                                                             G1 14530
      GO TO 118
  121 CONTINUE
                                                                              G1 14540
  119 IF(P$(2,C+Y).EQ.1)GO TO 122
                                                                             G1 14550
      IF(INPTCV(C).GT.1)GD TO 118
                                                                             G1 14560
      DC 123 J=1, PCINTL
                                                                             G1 14570
      Z = (LISTL(J)-1)*N2
                                                                              G1 14580
      IF(P$(2, Z+C).NE.1)GO TC 123
                                                                             G1 14590
      ESSIS(LISTL(J)) = FSSIS(LISTL(J))+1
                                                                             G1 14600
      GD TO 118
                                                                             G1 14517
                                                                             G1 14620
  123 CONTINUE
  122 \text{ INPTCV(C)} = \text{INPTCV(C)} + 1
                                                                             G1 14630
  118 CONTINUE
                                                                             G1 14640
                                                                              G1 14650
C
C
      SECTION 24, IS LISTL EMPTY ?
                                                                             G1 14560
C
                                                                             G1 14679
   24 IF(PCINTL.EG.0)GO TO 32
                                                                             G1 14680
C
      IF NOT EMPTY, GO TO SECTION 29
                                                                             G1 14690
C
                                                                             G1 14700
C
      SECTION 29, RETROER LISTL
                                                                             G1 14710
C
                                                                             G1 14720
   29 CALL FLANDC(GCD)
                                                                             G1 14730
C
                                                                             G1 14740
C
      SECTION 31, REMOVE P FROM LISTO
                                                                             G1 14750
C
                                                                             G1 14760
   31 POINTC = POINTC - 1
                                                                             G1 14770
      IF(POINTC.LE.O) GO TO 131
                                                                             G1 14780
      DO 129 I=1, POINTO
                                                                             G1 14790
      IF(LISTC(I).NE.P)GOT0129
                                                                             G1 14800
      COMPRESS LISTC, OVERWRITING ENTRY P
                                                                             61 14810
      DO 130 J=I,PCINTC
                                                                             G1 14820
      LISTC(J) = LISTC(J+1)
                                                                             G1 14830
  130 CONTINUE
                                                                             61 14840
      GO TO 131
                                                                             G1 14850
  129 CONTINUE
                                                                             G1 14860
  131 CONTINUE
                                                                             G1 14870
      RETURN TO SECTION 18
                                                                             G1 14880
```

```
GO TO 18
                                                                              G1 14890
                                                                              G1 14900
C
Ċ
      SECTION 21, ARE WE AT BOTTOM OF LISTL ?
                                                                              G1 14910
C
                                                                             31 14920
   21 IF( PCINTL .EQ. SUBL ) GO TO 25
                                                                             G1 14930
                                                                              G1 14940
      GO TO 22
                                                                             G1 14950
C
C
      SECTION 22, SELECT NEXT LOWER ELEMENT IN LISTL AS NEW TH
                                                                             G1 14960
C
                                                                              G1 14970
   22 SUBL = SUBL + 1
                                                                              G1 14980
      TH = LISTL(SUBL)
                                                                              G1 14990
      GD TO 19
                                                                              G1 15000
                                                                              G1 15010
C
      SECTION 25, SEARCH LISTC FOR EFFECTIVELY CONNECTIBLE FN., "EFFCCN"G1 15020
                                                                              G1 15030
   25 IF(PDINTC.EQ.O)GO TO 32
                                                                              G1 15040
      IF(VF$UB1.EQ.O) GOTO32
                                                                              G1 15050
      COUNT = 0
                                                                              G1 15060
  137 IFICOUNT.EQ.POINTCIGO TO 138
                                                                              G1 15070
      COUNT = COUNT + 1
                                                                              G1 15080
                                                                              31 15090
      CCC = LISTC(COUNT)
                                                                              G1 15100
      X = (CCO - 1) * N2
      00 139 I=1, VF$UB1
                                                                              G1 15110
      IF(P$(2,VF$1(I)+X).EQ.1)GO TO 140
                                                                              G1 15120
  139 CONTINUE
                                                                              G1 15130
      GO TO 137
                                                                              G1 15140
  140 CONTINUE
                                                                              G1 15150
                                                                              G1 15160
      DOES COD COVER ANY ESSENTIAL ONES ?
                                                                              G1 15170
      DO 141 I=1, VF$UB1
      IF (INPTCV(VF$1(I)).NE.1)GC TO 141
                                                                              G1 15180
      IF(P$(2,VF$1(I)+X).EQ.1)GO TO 142
                                                                              G1 15190
                                                                             G1 15200
  141 CONTINUE
      G0 T0137
                                                                              G1 15210
  142 CONTINUE
                                                                              G1 15220
      CENNECT CCO
                                                                              G1 15230
      EFFCON = CCO
                                                                              G1 15240
                                                                              G1 15250
      GO TO 27
                                                                              31 15260
  138 CONTINUE
      IF HERE, NO ESSENTIAL 1'S COULD BE COVERED - SO CONNECT EFFECTIVE G1 15270
         CONNECTIBLE ONE AT A TIME UNTIL NO MORE CAN BE ACCED. THEN GO G1 15280
Ċ
         TO SECTION 32 (SINCE NO MORE ORIGINAL INPUTS CAN BE ELIMINATED.G1 15290
      O = TVUCO
                                                                              G1 15300
  144 IF(COUNT.EQ.POINTC)GO TO 32
                                                                              G1 15313
                                                                              G1 15320
      COUNT = COUNT + 1
      CCO = LISTC(COUNT)
                                                                              G1 15330
      X = (CCO-1) * N2
                                                                              G1 15340
      DO 145 I=1, VF$UB1
                                                                              G1 15350
                                                                              G1 15360
      IF(P$(2, VF$1(I)+X).EQ.1)GC TO 146
                                                                              G1 15370
  145 CONTINUE
      GO TO 144
                                                                              G1 15380
                                                                              G1 15390
  146 CONTINUE
                                                                              G1 15400
      CONNECT CCG, UPDATE ARRAYS (INCLUDING VF$1)
C
                                                                              G1 15410
      CALL CONCOCIGOO, COO!
                                                                              G1 15420
      GO TO 144
                                                                              G1 15430
                                                                              G1 15440
      SECTIONS 27 AND 28 - CONNECT EFFCON AND UPDATE ARRAYS
                                                                              31 15450
                                                                             G1 15460
      UPDATE ESSIS
   27 \times = (CCO-1)*N2
                                                                              G1 1547(
      00 155 II=1,F$UB1
                                                                              G1 15480
      I = F$1(II)
                                                                              G1 15490
```

```
G1 15500
      IF(P$(2,X+I).NE.1)GO TO 155
                                                                             G1 15510
      IF (INPTCV(I).NE.1)GO TO 155
      IF HERE, SOME GATE WILL LOSE AN ESSENTIAL ONE
                                                                             G1 15520
C
                                                                            G1 15530
      Y = LIPRED(GCO)
                                                                             G1 15540
      DO 156 J=1,Y
                                                                             G1 15550
      XY = IPRED(J,GCO)
      Z = (XY-1) * N2
                                                                             31 15560
                                                                             G1 15570
      IF(P$(2,Z+I).NE.1)GO TO 156
      ESSIS(XY) = ESSIS(XY) - 1
                                                                             G1 15580
      GO TO 155
                                                                             G1 15590
  156 CONTINUE
                                                                             G1 15600
                                                                            G1 15610
  155 CONTINUE
                                                                             G1 15620
      CALL CONCCO(GCO, EFFCON)
      ALTHOUGH SOME ARRAYS ARE UPDATED BY PREVIOUS CALL, WE MUST ALSO
                                                                             G1 15630
C
C
         UPDATE:
                   ESSIS, LISTL, PCINTL, LISTC, POINTC
                                                                             G1 15640
C
                                                                            G1 15650
      UPDATE LISTC, POINTC
      Y = POINTC
                                                                             G1 15660
      DO 152 I=1,Y
                                                                            G1 15670
                                                                             G1 15680
      IF(LISTC(I).NE.CCO)GO TO 152
      POINTC = POINTC - 1
                                                                             G1 15690
      IF(I.GT.POINTC) GO TO 153
                                                                            G1 15700
      DO 154 J=I, POINTC
                                                                            G1 15710
      LISTC(J) = LISTC(J+1)
                                                                            31 15720
  154 CONTINUE
                                                                            G1 15730
  152 CONTINUE
                                                                            G1 15740
  153 CONTINUE
                                                                            G1 15750
      UPDATE LISTL, POINTL
                                                                             G1 15760
                                                                            G1 15770
      CALL ORDERL (GCO)
      GO TO 18
                                                                            G1 15780
                                                                            G1 15790
      SECTION 32 - UPDATE GSMALL'S FOR THOSE GATES STILL CONNECTED
                                                                             G1 15800
C
         TO GCO
                                                                            31 15810
C
                                                                             G1 15820
C
                                                                            G1 15830
      FORCE NECESSARY ZEROS IN GSMALL
                                                                            G1 15840
   32 \times = LIPPED(GCO)
      DO 157 J=1,X
                                                                            G1 15850
      Y = IPRED(J_*GCO)
                                                                            G1 15860
      DJ 158 II=1,F$UBO
                                                                            G1 15870
      I = F\$O(II)
                                                                            G1 15880
      GSMALL(Y,I) = -100
                                                                            G1 15890
  158 CCNTINUE
                                                                            G1 15900
  157 CONTINUE
                                                                            G1 15910
C
      CHOOSE NECESSARY ONES ACCORDING TO RORDER
                                                                            G1 15920
      POINT = 0
                                                                            G1 15930
      DO 170 I = 1.NR
                                                                            G1 15940
      Y = RORDER(I)
                                                                            G1 15950
      IF(INC $MX(Y,GCO).EQ.O) GO TO 170
                                                                            G1 15960
      POINT = POINT + 1
                                                                            G1 15970
      ORDERP(POINT) = Y
                                                                            31 15980
  170 CONTINUE
                                                                            G1 15990
C
      NOTE NOW POINT = LIPRED(GCO)
                                                                            G1 16000
      DO 159 KK=1,F$UB1
                                                                            G1 16010
      K = F$1(KK)
                                                                            G1 16020
      GOTO(11111,22222,33333,44444,55555),JFLAG
                                                                            G1 16030
      VERSION 1: 1) NEW
                              2) OLD
                                                                            G1 16040
111111 PICK1 = 0
                                                                            G1 16050
      PICK2 = 0
                                                                            31 16060
      DO 173 J=1, POINT
                                                                            G1 16070
      Y = ORDERP(J)
                                                                            G1 16080
      IF(P$(2,(Y-1)*N2+K).NE.1) GD TO 173
                                                                            G1 16090
      IF(GSMALL(Y,K).GE.1) GO TO 159
                                                                            G1 16100
```

```
IF(PICK1.NE.O) GO TO 173
                                                                               G1 16110
      IF(PICK2.EQ.0) PICK2 = Y
                                                                               G1 16120
      IF(ORIGIN(Y).EQ.O) PICK1 = Y
                                                                               G1 16130
  173 CONTINUE
                                                                               31 16140
      IF(PICK1.EQ.O) GO TO 188
                                                                               G1 16150
      GSMALL(PICK1,K) = 1
                                                                               G1 16160
      GO TO 159
                                                                               G1 16170
  188 \text{ GSMALL(PICK2,K)} = 1
                                                                               G1 16180
      GD TO 159
                                                                               G1 16190
      VERSION 2 : 1) NG + NEV
                                     21 OEV
                                               31 DG
                                                                               G1 16200
22222 PICK1 = 0
                                                                               G1 16210
      PICK2 = 0
                                                                               51 16220
      PICK3 = 0
                                                                               G1 16230
      90 175 J = 1, POINT
                                                                               G1 16240
      Y = ORDERP(J)
                                                                               31 16250
      IF(P$(2,(Y-1)*N2+K).NE.1) GO TO 175
                                                                               G1 16260
      IF (GSMALL(Y,K).GE.1) GO TO 159
                                                                               31 16270
                                                                               G1 16280
      IF(PICK1.NE.O) GO TO 175
                                                                               G1 16290
      IF(ORIGIN(Y).NE.O) GO TO 207
      PICK1 = Y
                                                                               31 16300
      GJ TO 175
                                                                               G1 16310
  207 IF(PICK2.NE.O) GO TO 175
                                                                               G1 16320
      IF(Y.GT.N) GO TO 208
                                                                               G1 16330
      PICK2 = Y
                                                                               G1 16340
                                                                               G1 16350
      GC TO 175
  208 IF(PICK3.E0.0) PICK3 = Y
                                                                               G1 16360
  175 CONTINUE
                                                                               G1 16370
                                                                               G1 16380
      IF(PICK1.EQ.0) GD TO 176
      GSMALL(PICK1,K) = 1
                                                                               G1 16390
      SD TO 159
                                                                               G1 16400
  176 IF(PICK2.EQ.O) GO TO 209
                                                                               G1 16410
      GSMALL(PICK2,K) = 1
                                                                               G1 16420
                                                                               G1 16430
      GO TO 159
                                                                               G1 16440
  209 \text{ GSMALL}(PICK3,K) = 1
      GO TO 159
                                                                               G1 16450
                                21 CEV
C
      VERSION 3 :
                    1) NEV
                                          3) NG
                                                   41 DG
                                                                               G1 16460
33333 PICK1 = 0
                                                                               31 16470
      PICK2 = 0
                                                                               G1 16480
                                                                               31 16490
      PICK3 = 0
                                                                               G1 16500
      PICK4 = 0
      DO 177 J=1, POINT
                                                                               G1 16510
      Y = QRDERP(PCINT+1-J)
                                                                               G1 16520
      IF(P$(2,(Y-1)*N2+K).NE.1) GO TO 177
                                                                               G1 16530
      TF(GSMALL(Y.K).GE.1) GO TO 159
                                                                               G1 16540
      IF(Y.GT.N) GO TO 178
                                                                               G1 16550
      IF(ORIGIN(Y).NE.O) GO TO 179
                                                                               G1 16560
                                                                               G1 16570
      PICK1 = Y
      GO TO 177
                                                                               G1 16580
  179 PICK2 = Y
                                                                               G1 16590
      GO TO 177
                                                                               G1 16600
                                                                               G1 16610
  178 IF (ORIGIN(Y).NE.O) GO TO 210
                                                                               G1 16620
      PICK3 = Y
                                                                               G1 16630
      GO TO 177
                                                                               31 16640
  210 PICK4 = Y
  177 CONTINUE
                                                                               G1 16650
                                                                               31 16660
      CHOICE = PICK1
                                                                               31 16670
      IF (CHOICE.NE.O) GO TO 211
      CHOICE = PICK2
                                                                               G1 16680
                                                                               31 16690
      IF (CHOICE.NE.O) GO TO 211
                                                                               31 16700
      CHOICE = PICK3
                                                                               G1 16710
      IF (CHOICE.NE.O) GO TO 211
```

```
CHOICE = PICK4
                                                                             G1 16720
  211 GSMALL(CHOICE, K) = 1
                                                                             G1 16730
                                                                             G1 16740
      GO TO 159
                               21 NG
      VERSION 4:
                     1) NEV
                                       3) DEV
                                                 41 OG
                                                                             G1 16750
44444 PICK1 = 0
                                                                             G1 16760
      PICK2 = 0
                                                                             31 16770
      PICK3 = 0
                                                                             G1 16780
      PICK4 = 0
                                                                             G1 16790
      DO 180 J = 1, POINT
                                                                             31 16800
      Y = ORDERP(POINT+1-J)
                                                                             G1 16810
      IF(P$(2,(Y-1)*N2+K).NE.1) GO TO 180
                                                                             G1 16820
      IF (GSMALL(Y,K).GE.1) GO TO 159
                                                                             S1 16830
      IF(Y.GT.N) GD TO 181
                                                                             G1 16840
      IF(ORIGIN(Y).NE.O) GO TO 182
                                                                             G1 16850
      PICK1 = Y
                                                                             31 16860
      GO TO 180
                                                                             G1 16870
  182 PICK3 = Y
                                                                             G1 16880
      GO TO 180
                                                                             G1 16890
  181 IF(CRIGIN(Y).NE.O) GO TO 212
                                                                             G1 16930
      PICK2 = Y
                                                                             31 16910
      GO TO 180
                                                                             G1 16920
  212 PICK4 = Y
                                                                             G1 16930
  180 CONTINUE
                                                                             31 16940
      CHOICE = PICK1
                                                                             G1 16950
      IF (CHOICE . NE . O) GO TO
                              213
                                                                             G1 16960
      CHOICE = PICK2
                                                                             G1 16970
      IF(CHOICE.NE.O) GO TO 213
                                                                             G1 16980
      CHOICE = PICK3
                                                                             G1 16990
      IF(CHOICE.NE.O) GO TO 213
                                                                             31 17000
      CHOICE = PICK4
                                                                             G1 17010
  213 GSMALL(CHOICE,K) = 1
                                                                             G1 17020
      GO TO 159
                                                                             G1 17030
      VERSION 5 :
                     1) EV
                             2) G
                                                                             G1 17040
55555 PICK1 = 0
                                                                             31 17050
      PICK2 = 0
                                                                             G1 17060
      DO 183 J = 1. POINT
                                                                             G1 17070
      Y = ORDERP(POINT+1-J)
                                                                             G1 17080
      IF (P$(2, (Y-1)*N2+K).NE.1) GO TO 183
                                                                             G1 17090
      IF(GSMALL(Y,K).GE.1) GO TO 159
                                                                             G1 17100
      PICK2 = Y
                                                                             31 17110
      IF(Y.LE.N)PICK1 = Y
                                                                             G1 17120
  183 CONTINUE
                                                                             G1 17130
      CHOICE = PICK2
                                                                             31 17140
      IF(PICK1.NE.O) CHOICE = PICK1
                                                                             G1 17150
      GSMALL(CHOICE_*K) = 1
                                                                             G1 17160
      GO TO 159
                                                                             G1 17170
  159 CONTINUE
                                                                             G1 17180
      GO TO 5
                                                                             G1 17190
                                                                             G1 17200
C
      SECTION 7 - TRY TO REMOVE CONNECTIONS PREVIOUSLY ADDED
                                                                             G1 17210
C
                                                                             31 17220
    7 CALL SUBNET
                                                                             G1 17230
      CALL PVALUE
                                                                             G1 17240
      CALL MINI2 (IMPROV)
                                                                             G1 17250
C
      CALCULATE NUMBERS OF GATES AND CONNECTIONS IN NEW NETWORK
                                                                             G1 17260
      GAFTER = M
                                                                             G1 17270
      CAFTER = 0
                                                                             G1 17280
      DO 165 I=1,NR
                                                                             G1 17290
      CAFTER = CAFTER + LISUCC(I)
                                                                             G1 17300
      IF(I.LE.NM) GO TO 165
                                                                             G1 17310
      IF(LISUCC(I).GT.O) GAFTER = GAFTER + 1
                                                                             G1 17320
```

```
165 CONTINUE
                                                                          G1 17330
C
      COST OF NEW NETWORK IS :
                                                                          G1 17340
      NEWCST = GAFTEP*A + CAFTER*B
                                                                          G1 17350
      COST OF ORIGINAL NETWORK WAS :
                                                                          G1 17360
C
      CLDCST = GBEFCR*A + CBEFOR*B
                                                                          G1 17370
C
      SO COST SAVINGS OF TRANSFORMED NETWORK IS:
                                                                          G1 17380
      SAVING = OLDCST - NEWCST
                                                                          G1 17390
      PRINT 8963, SAVING
                                                                          G1 17400
 8963 FORMAT( ! SAVING = 1,15)
                                                                          G1 17410
C
                                                                          31 17420
C
      SECTION 8 - IMPROVED NETWORK OBTAINED ?
                                                                          G1 17430
C
                                                                          G1 17440
    8 IF(SAVING.GE.O)GO TO 9
                                                                          G1 17450
      GO TO 14
                                                                          G1 17460
                                                                          G1 17470
C
C
      SECTION 9 - IMPROVED NETWORK, SO UPDATE ARKAYS
                                                                          G1 17486
                                                                          G1 17490
Ċ
      UPDATE COST
                                                                          G1 17500
     COST = COST - SAVING
                                                                          G1 17510
    9 GO TO 10
                                                                          G1 17520
C
                                                                          G1 17530
                                                                          G1 17540
C
      SECTION 14 - RESTORE ORIGINAL NETWORK
                                                                          G1 17550
      RESTORE INC $MX FROM QINC $M
                                                                          G1 17560
   14 DO 171 I=1,NR
                                                                          G1 17570
      DO 171 J=1.NR
                                                                          G1 17580
  171 INCSMX(I.J) = CINCSM(I.J)
                                                                          31 17590
      CALL SUBNET
                                                                          G1 17600
      CALL PVALUE
                                                                          31 17610
                                                                          G1 17620
      SD TO 10
C
                                                                          G1 17630
C
      SECTION 10 - END OF PROCEDURE II (PROCII)
                                                                          G1 17640
C
                                                                          G1 17650
   10 RETURN
                                                                          G1 17660
      END.
                                                                          31 17670
      SUBROUTINE RNONES(GCO.TH.
                                      FLAG)
                                                                          G1 17680
C
      G1 17700
      THIS SUBROUTINE PERFORMS NECESSARY UPDATES WHEN AN INPUT TH IS
                                                                          G1 17710
         REMOVED FROM A GATE GCO. ARRAY "ESSIS" IS ONLY PARTIALLY
                                                                          G1 17720
         UPDATED (SO MUST BE COMPLETED BY CALLING SECTION OF PROGRAM).
                                                                          G1 17730
         (RNONES = REMOVE NON-ESSENTIAL INPUT)
                                                                          G1 17740
                                                                          G1 17750
      IF FLAG = 0, SKIP UPDATE OF LISTL, POINTL, INPTCV
                                                                          G1 17760
      IF FLAG=1, SKIP LISTL UPDATE, IF FLAG=2, DO NOT SKIP
                                                                          G1 17770
                                                                          G1 17780
C
                                                                          G1 17790
      IMPLICIT INTEGER *4(4-T, V-Z, $), REAL(U)
C
                                                                          31 17800
      DEFINITIONS OF "CCMMCN" VARIABLES CAN BE FOUND IN MAIN PROGRAM.
                                                                          G1 17810
C
                                                                          G1 17820
      I, J, X ARE USED AS JUST TEMPORARY VAPIABLES.
                                                                          G1 17830
                                                                          G1 17840
      COMMON NEPMAX
                                                                          G1 17850
                                                                          31 17860
      NOMMOD
               M
                                             Α
                                                            , B
                                                                          G1 17870
               R
                            , N2
                                            , N1
                                                           , NR
     1
               NM
                            , KFLAG
                                            . JFLAG
                                                            , COST
                                                                          31 17880
     2
               LEVM
                            • NRN2
                                                                          G1 17890
     3
                                                           , NN2
                                            • NM1
                                                                         G1 17900
      COMMON
               ISUCC(40,40) , LISUCC(40)
                                          , IPRED(40,40) , LIPRED(40)
               INC$MX(40,40), SUC$MX(40,40), P$(2,1280) , UNAME(40)
                                                                          G1 17910
```

```
. LGLIST(40)
                                              , HLIST (40,40) , TIME
                                                                               G1 17920
              GLEVEL (40)
    COMMON
                                              , S
                                                               , RSCONN(100) G1 17930

    RTCCNN(100)

                             . POINTA
                                                               .F$1(32)
    COMMON
              IFLAG
                                              · ESS1S(40)
                                                                              G1 17940
                             ,INPTCV(32)
                                              ,LISTC(40)
                                                               , POINTC
                                                                               G1 17950
          ,F$UB1
                             , POINTL
                                              ,ORIGIN(40)
                                                               , IPATH(40)
                                                                               G1 17960
          ·LISTL(40)
   2
                             ,VF$1(32)
          , POINTR
                                              .VF$UB1
                                                               .GSMALL(40,32)G1 17970
   3
                                              , LPOTAB (40)
    COMMON
             POTAB (200, 42), PPOTAB (40)
                                                               ·NRPLC(2)
                                                                               G1 17980
                                              , IDX0E(32)
                             , IDX0(32)
                                                                              G1 17990
          , RPLC (2, 40)
                                                               , IDX1(32)
                             .SUMP(32)
                                              ,SETT1(32)
          .IDX1E(32)
                                                               . NOT1
                                                                              G1 18000
          , SETS1(40)
                             , NOSI
                                              , SETS (40)
                                                               . NOS
                                                                               G1 18010
                             ,SUMS 2 (32)
                                              , SETS2 (200)
                                                               ,NOS2
                                                                              31 18020
          , STS
   5
          , LIP
                                              , KEYA
                                                               , KEYB
                                                                               G1 18030
                             , NOOE
   6
          , NO0
                             ,NO1
                                              , NO1E
                                                               ,$GT
                                                                               31 18040
          , SLTH
                             . $ PH
                                              . $NOE
                                                               , GI
                                                                              G1 18050
                             NOT1SV
                                             . NOSISV
                                                              , LMTS2
                                                                              G1 18060
    COMMON
    UPDATE INC $MX
                                                                              G1 18070
    INC $MX (TH.GCO) =0
                                                                              G1 18080
    UPDATE LISUCC AND ISUCC
                                                                              G1 18090
                                                                              G1 18100
    X = LISUCC(TH)
                                                                              G1 18110
    DO 93 I=1.X
    IF(ISUCC(I,TH).EQ.GCO)GOTO94
                                                                              G1 18120
 93 CONTINUE
                                                                              G1 18130
 94 \times = LISUCC(TH) - 1
                                                                              G1 18140
    LISUCC(TH) = X
                                                                              G1 18150
                                                                              G1 18160
    IF(I.GT.X)GOTO129
                                                                              G1 18170
    DO 95 J=I.X
    ISUCC(J,TH)=ISUCC(J+1,TH)
                                                                              G1 18180
 95 CONTINUE
                                                                              G1 18190
    UPDATE LIPRED AND IPRED
                                                                              G1 18200
129 X = LIPRED(GCO)
                                                                              G1 18210
    DO 98 I=1,X
                                                                              G1 18220
    IF (IPR EC(I, GCO). EQ. TH) GO TO
                                                                              G1 18230
 98 CONTINUE
                                                                              G1 18240
                                                                              G1 18250
  8 X = X - 1
    LIPRED(GCO) = LIPRED(GCO) - 1
                                                                              G1 18260
    IF(I.GT.X)GOTO92
                                                                              G1 18270
    D7 99 J=I.X
                                                                              G1 18280
    IPRED(J,GCO)=IPRED(J+1,GCO)
                                                                              G1 18290
                                                                              G1 18300
G1 18310
 99 CONTINUE
    UPDATE SUC $MX
 92 CALL SUCCES
                                                                              G1 18320
    IF(FLAG.EQ.O)GOTO177
                                                                              G1 18330
    UPCATE LISTL AND POINTL
                                                                              G1 18340
    IF (FLAG.EQ.1)GOTO128
                                                                              31 18350
    DO 105 I=1, POINTL
                                                                              G1 18360
    IF(LISTL(I).EQ.TH)GO TO 107
                                                                              G1 18370
105 CONTINUE
                                                                              G1 18380
107 POINTL= POINTL - 1
                                                                              G1 18390
    DO 108 J=I, POINTL
                                                                              31 18400
    LISTL(J)=LISTL(J+1)
                                                                              G1 18410
108 CONTINUE
                                                                              G1 18420
    UPDATE INPTCV
                                                                              G1 18430
128 X = (TH-1)*N2
                                                                              G1 18440
    DO 110 I=1,F$UB1
                                                                              GI 18450
    IF (P$(2,X+F$1(I)).LE.0)GOT 0110
                                                                              31 18460
    INPTCV(F$1(I))=INPTCV(F$1(I))-1
                                                                              G1 18470
110 CONTINUE
                                                                              31 18480
    RECORD THE DISCONNECTION
                                                                              G1 18490
177 RETURN
                                                                              G1 18500
    END
                                                                              G1 18510
```

C

C

C

C

C

C

```
SUBROUTINE SUBNET
                                                                                    G1 18520
       IMPLICIT INTEGER*4(A-T.V-Z.$). REAL(U)
                                                                                    G1 18530
\cap
                                                                                    G1 18540
Ċ
       DEFINITIONS OF 'COMMON' VARIABLES CAN BE FOUND IN MAIN PROGRAM.
                                                                                    G1 18550
C
                                                                                    G1 18560
       COMMON NEPMAX
                                                                                    G1 18570
       CCMMON
                 N
                                   М
                                                    Α
                                                                      В
                                                                                    G1 18580
                                                  9
                 R
      1
                                   N2
                                                    NI
                                                                      NR
                                                                                    G1 18590
                 1. M
      2
                                   KFLAG
                                                    JELAG
                                                                     COST
                                                                                    G1 18600
      3
                 LEVM
                                   NRN2
                                                    NM1
                                                                      NN<sub>2</sub>
                                                                                    G1 18610
       COMMON
                 ISUCC(40,40)
                                   LISUCC(40)
                                                    IPRED(40,40)
                                                                      LIPREC(40)
                                                                                    G1 18620
                                                  9
      1
                 INC$MX(40,40),
                                   SUC$MX(40,40),
                                                    P$(2,1280)
                                                                      UNAME (40)
                                                                                    G1 18630
      2
                 GLEVEL(40)
                                   LGLIST(40)
                                                   HLIST (40,40)
                                                                      TIME
                                                                                    G1 18640
       NOMPOS
                                                  , S
                                                                                    G1 18650
                 Ŧ
                                 , RTCCNN(100)
                                                                    , RSCONN(100)
      COMMON
                                                  , ESSIS(40)
                                                                    ,F$1(32)
                 IFLAG
                                 . POINTA
                                                                                    31 18660
                                                                    , POINTC
      1
             .F$UB1
                                 ,INPTCV(32)
                                                  ·LISTC(40)
                                                                                    G1 18670
      2
                                                                    , IPATH(40)
                                                                                    G1 18680
             ·LISTL(40)
                                 , POINTL
                                                  , ORIGIN(40)
                                                  , VF$UB1
      3
             , POINTR
                                 , VF$1(32)
                                                                    ,GSMALL(40,32)G1 18690
      СОММОИ
                POTAB(200,42),PPOTAB(40)
                                                  · LPOTAB (40)
                                                                    .NRPLC(2)
                                                                                    G1 18700
             , RPLC (2,40)
                                 , IDX0(32)
                                                  . IDXOE(32)
                                                                    , IDX1(32)
                                                                                    G1 18710
      1
      2
             , IDX1E(32)
                                 ,SUMP(32)
                                                  ,SETT1(32)
                                                                    , NOTI
                                                                                    G1 18720
             ·SETS1(40)
                                                                    , NOS
      3
                                 .NOS1
                                                  , SETS (40)
                                                                                    G1 18730
                                                  , SETS 2 ( 200)
                                                                    , NOS2
      4
             ,STS
                                 ,SUMS2(32)
                                                                                    G1 18740
      5
                                 .NOOE
             , LIP
                                                                                    G1 18750
                                                  , KEYA
                                                                    , KEYB
     6
             .NG0
                                 .NO1
                                                  ,NOIE
                                                                    , SGT
                                                                                    31 18760
      7
             . SLTH
                                 . $PW
                                                  , $NOE
                                                                    , G$$$$$
                                                                                    G1 18770
                                NOTISV
       NOMMED
                                                 , NOSISV
                                                                                    G1 18780
                                                                   ,LMTS2
       DIMENSION X(40), LX(40,2), OUTO(40)
                                                                                    G1 18790
0
       ENTRY PRESUC
                                                                                    G1 18800
    1 CONTINUE
                                                                                    G1 18810
       DO 10 GI=1, NR
                                                                                    31 18820
        LS=0
                                                                                    G1 18830
        LP=0
                                                                                    G1 18840
        02 5 GJ=1,NR
                                                                                    31 18850
         IF(INC $MX(GI,GJ).EQ.O) GO TO 3
                                                                                    G1 18860
          LS = LS + 1
                                                                                    G1 18870
          ISUCC(LS,GI)=GJ
                                                                                    31 18880
                                                                                    G1 18890
          GO TO 5
         IF(INC$MX(GJ,GI).EQ.O) GO TO 5
                                                                                    G1 18900
          LP = LP + 1
                                                                                    G1 18910
          IPRED(LP,GI)=GJ
                                                                                    G1 18920
        CONTINUE
                                                                                    31 18930
        LISUCC(GI)=LS
                                                                                    51 18940
        LIPRED(GI)=LP
                                                                                    G1 18950
   10 CONTINUE
                                                                                    31 18960
                                                                                    G1 18970
       ENTRY SUCCES
                                                                                    G1 18980
       DO 21 GI=1, NR
                                                                                    G1 18990
       00 21 GJ=1,NR
                                                                                    G1 19000
        SUC $MX(GI,GJ)=0
                                                                                    G1 19010
   21 CONTINUE
                                                                                    G1 19020
       DO 30 GJ=N1.NR
                                                                                    G1 19030
       DD 22 GS=1,NR
                                                                                    G1 19040
                                                                                    31 19050
         X(GS) = 0
   22
       CONTINUE
                                                                                    G1 19060
        X(GJ)=1
                                                                                    G1 19070
        L \Omega = 1
                                                                                    G1 19080
                                                                                    G1 19090
        LX(1,1)=GJ
                                                                                    G1 19100
        V = 1
                                                                                    G1 19110
   23
       CONTINUE
```

```
G1 19120
       V=1-V
                                                                              G1 19130
       SW0=1+V
                                                                               G1 19140
       SW1 = 2-V
                                                                               GI 19150
       L1 = 0
       DO 28 LL=1,LO
                                                                              G1 19160
                                                                               G1 19170
        GM=LX(LL.SWO)
                                                                               G1 19180
        LIP=LIPRED(GM)
                                                                               G1 19190
        IF(LIP.EQ.0) GO TO 28
                                                                              G1 19200
        00 26 LP=1.LIP
         GP=IPRED(LP,GM)
                                                                               G1 19210
         IF(X(GP)-GT-0) GO TO 26
                                                                               31 19220
                                                                               G1 19230
          SUC $MX(GP,GJ)=1
                                                                               G1 19240
          L1=L1+1
                                                                              G1 19250
          LX(L1,SW1)=GP
                                                                              G1 19260
          X(GP)=1
   26
        CONTINUE
                                                                              G1 19270
       CONTINUE
                                                                               G1 19280
   28
       IF(L1.EQ.0) GO TO 30
                                                                               G1 19290
                                                                               G1 19300
       L0=L1
       GO TO 23
                                                                              G1 19310
   30 CONTINUE
                                                                               G1 19320
                                                                              G1 19330
C
C
      ENTRY LEVEL
                                                                               G1 19340
      DO 40 GJ=1,NR
                                                                              G1 19350
       OUTO(GJ)=LISUCC(GJ)
                                                                               31 19360
                                                                              G1 19370
       GLEVEL(GJ)=-1
       CONT INUE
                                                                              G1 19380
      LEV=0
                                                                              G1 19390
   45 LEV=LEV+1
                                                                              G1 19400
      G=0
                                                                              G1 19410
      DO 50 GJ=1,NR
                                                                              G1 19420
       IF(OUTO(GJ).GT.O .OR. GLEVEL(GJ).GT.O) GO TO 50
                                                                              G1 19430
                                                                              G1 19440
       HLIST(G, LEV) = GJ
                                                                              G1 19450
       GLEVEL (GJ)=LEV
                                                                              G1 19460
   50 CONTINUE
                                                                              31 19470
      IF(G.EQ.O) RETURN
                                                                              G1 19480
      LGLIST(LEV)=G
                                                                              G1 19490
      DO 60 GG=1,G
                                                                              31 19500
       GJ=HLIST(GG, LEV)
                                                                              G1 19510
       LIP=LIPRED(GJ)
                                                                              G1 19520
                                                                              G1 19530
       IF(LIP.EQ.O) GO TO 60
                                                                              G1 19540
       DO 55 LP=1,LIP
        GP=IPRED(LP.GJ)
                                                                              G1 19550
        QUTO(GP) = QUTO(GP)-1
                                                                              G1 19560
   55 CONTINUE
                                                                              G1 19570
   60 CONTINUE
                                                                              G1 19580
      LEVM=LEV
                                                                              G1 19590
      GO TO 45
                                                                              G1 19600
C
                                                                              G1 19610
C
                                                                              G1 19620
C
                                                                              G1 19630
      ENTRY PVALUE
                                                                              G1 19640
      DO 100 L=NN2, NRN2
                                                                              G1 19650
       P$(1,L)=1
                                                                              G1 19660
  100 CONTINUE
                                                                              31 19670
C
                                                                              G1 19680
      LEV=LEVM
                                                                              G1 19690
  110 CONTINUE
                                                                              G1 19700
      LO=LGLIST(LEV)
                                                                              Gl 19710
      90 130 L=1,L0
                                                                              G1 19720
```

```
G1 19730
       GI=HLIST(L.LEV)
       LIS=LISUCC(GI)
                                                                                G1 19740
       BSGI=(GI-1)*N2
                                                                                31 19750
       LJTH=0
                                                                                G1 19760
       DC 115 JTH=1,N2
                                                                                G1 19770
        IF(P$(1,8SGI+JTH).EQ.0) GO TO 115
                                                                                31 19780
        LJTH=LJTH+1
                                                                                G1 19790
                                                                                31 19800
        HTL=(HTLJ)X
  115 CONTINUE
                                                                                G1 19810
       IF(LJTH.EQ.O) GO TO 130
                                                                                G1 19820
       DO 125 LS=1.LIS
                                                                                G1 19830
        CS=ISUCC(LS,GI)
                                                                                G1 19840
        BSGS = (GS - 1) * N2
                                                                                G1 19850
        DO 120 LJ=1, LJTH
                                                                                G1 19860
         P$(1,X(LJ)+BSGS)=0
                                                                                G1 19870
        CONTINUE
  120
                                                                                G1 19880
  125
       CONTINUE
                                                                                G1 19890
  130 CONTINUE
                                                                                G1 19900
      LEV=LEV-1
                                                                                G1 19910
      IF(LEV.GE.2) GO TO 110
                                                                                G1 19920
      RETURN
                                                                                G1 19930
(
                                                                                G1 19940
C
                                                                                G1 19950
C
                                                                                G1 19960
      ENTRY RSTRCT (KEYRST)
                                                                                G1 19970
      KEYRST = 0
                                                                                G1 19980
      IF(LEVM.GT.LMAX)GO TO 160
                                                                                G1 19990
      DO 150 GI=N1.NR
                                                                                G1 20000
       IF(LIPRED(GI).GT.FANIN)GO TO 160
                                                                                31 20010
       IF(LISUCC(GI).GT.FANOUT)GO TO 160
                                                                                G1 20020
  150 CONTINUE
                                                                                G1 20030
      RETURN
                                                                                G1 20040
  160 KEYRST=1
                                                                                G1 20050
      RETURN
                                                                                G1 20060
      ENTRY UNNECE
                                                                                G1 20070
C**** THIS ENTRY DISCONNECT ALL GATES FROM WHICH THERE IS NO PATH
                                                                                G1 20080
0
       TO OUTPUT GATES ****
                                                                                G1 20090
      T = T
                                                                                G1 20100
      DC 209 GI=NM1, NR
                                                                                G1 20110
       IF(GLEVEL(GI).EQ.1) GO TO 207
                                                                                31 20120
       DO 205 GJ=N1, NM
                                                                                G1 20130
                                                                                G1 20140
        IF(SUC$MX(GI,GJ).GT.O) GO TO 209
                                                                                G1 20150
  205
       CONTINUE
C**** GI IS REDUNDANT ****
                                                                                G1 20160
  207 CONTINUE
                                                                                G1 20170
       LIP=LIPRED(GI)
                                                                                G1 20180
       IF(LIP.EQ.O) GC TO 206
                                                                                G1 20190
       DC 203 LI=1,LIP
                                                                                G1 20200
                                                                                G1 20210
        GK=IPRED(LI,GI)
      IF(INC$MX(GK,GI).LE.O) GD TO 203
                                                                                G1 20220
        T = T + 1
                                                                                G1 20230
        RTCONN(T) = 100 * GK + GI
                                                                                G1 20240
        INC $MX(GK,GI)=0
                                                                                G1 20250
  203
       CONTINUE
                                                                                G1 20260
  206 LIS=LISUCC(GI)
                                                                                G1 20270
       IF(LIS.EQ.O) GC TO 209
                                                                                G1 20280
                                                                                G1 20290
       DO 204 LI=1, LIS
                                                                                G1 20300
        GK=ISUCC(LI.GI)
      IF(INC$MX(GI,GK).LE.O) GO TO 204
                                                                                G1 20310
        T = T + 1
                                                                                G1 20320
        RTCONN(T)=100*GI+GK
                                                                                G1 20330
```

INC \$ MX (GI, GK) = 0

204 CONTINUE

209 CONTINUE

IF (T.GT.TS) GO TO 1

RETURN

END

G 1	20340
G1	20350
G 1	20360
G1	20370
G1	20380
C 1	20390

```
00
C***
C
              PPPP
                          RRRR
                                       000
                                                  GGG
                                                             RRRR
Ċ
                   Р
                          R
                               R
                                     0
                                           0
                                                 G
                                                      G
                                                             R
                                                                  R
                                                                          AA
                                                                                           MM
              D
                                                             R
                                                                  R
                                                                                    М
                          R
                               R
                                     0
                                           0
                                                 G
                                                                         Δ
C
              oppp
                          RRRR
                                     0
                                           0
                                                 G
                                                     GG
                                                             RRRR
                                                                        AAAAA
                                                                                    M
              P
                          R
                              R
                                     0
                                           0
                                                 G
                                                      G
                                                             R
                                                                 R
                                                                        A
                                                                                    M
              ρ
                                                             R
                                                                  R
                                                                                            М
                                       000
                                                  GGG
                                                                         Δ
C
C
                                                                                     GGG
(
  N
              EEEEE
                          TTTTT
                                     TITIT
                                                 RRRR
                                                                A
                                                                                                 222
                             Ť
                                        T
                                                 R
                                                      R
                                                                                          G
                                                                                                2
C
  NN
        N
              E
                                                              A
                             T
                                                                                    G
              Ε
                                        T
                                                 R
                                                       R
                                                                                                    2
  N
     N
        NJ
                                                 RRRR
                                                                                    G
              EEE
                             7
                                        T
                                                                        XXXXX
                                                                                        GG
                                                                                                  2
       NN
                                                             AAAAA
C
                            T
                                        T
                                                     R
                                                             A
                                                                                    G
                                                                                          G
                                                                                                 2
        Ν
  N:
                                        T
                                                                                     GGG
                                                                                                22222
C
        N
              EEEEE
                            T
C
```

C

Ċ

C

C

CC

```
G2 00010
IMPLICIT INTEGER * 4(A-T, V-Z, $), REAL(U)
NOTE: ALL COMMON VARIBLES MIGHT NOT BE USED IN THIS PROGRAM.
                                                                  G2 00030
                                                                  G2 00040
COMMON VARIABLES:
                                                                  32 00050
   $37: POINTS TO A "COLUMN" OF POTAB. FOR EACH "ROW" THE ENTRY
                                                                  G2 00060
        IN THIS COL. TELLS GATE WHERE FN. IS REALIZED.
                                                                  G2
                                                                    00070
  $LTH: POINTS TO A "COLUMN" OF POTAB. FOR EACH "ROW" THE ENTRY
                                                                  G2 00080
        IN THIS COL. TELLS HOW MANY CONNECTIONS MUST BE ADDED.
                                                                  G2
                                                                     00090
  SNDE: POINTS TO A "COLUMN" OF POTAB. FOR EACH "ROW! THE ENTRY
                                                                  G2
                                                                     00100
        IN THIS CCL. TELLS THE NUMBER OF 1-ERRORS CREATED IF THIS G2
                                                                    00110
                                                                  G2 00120
        ROW IS USED.
   SPW: POINTS TO A "COLUMN" OF POTAB.
                                       FOR EACH 'ROW' THE ENTRY
                                                                  G2 00130
        IN THIS COLUMN TELLS THE PREFERENCE WEIGHT.
                                                                  G2 00140
     A: WEIGHT FOR NO. OF GATES IN COMPUTING COST FUNCTION.
                                                                  G2
                                                                     00150
     B: WEIGHT FOR NO. OF CONNECTIONS IN COMPUTING COST FUNCTION.
                                                                  G2 00160
  COST: COST OF NETWORK - A MEASURE OF NETWORK SIZE.
                                                                  G2 00170
 ESSIS: RECORDS NO. OF ESSENTIAL 1°S IN EVERY INPUT TO CURRENT GCOG2 00180
        (POSITIONS IN ESSIS CORRES. TO GATES NOT FEEDING GCO ARE
                                                                  G2 00190
        IGNORED).
                                                                  G2 00200
 FSUB1: POINTS TO LAST ELEMENT IN F$1.
                                                                  G2 00210
                                                                  G2 00220
   F$1: LISTS (CONSECUTIVELY) POSITIONS OF DESIRABLE 1'S (FOR
        COVERING) IN A CONNECTIBLE FUNCTION.
                                                                  G2
                                                                     00230
                                                                  G2 00240
    GI: LABEL OF A PAPTICULAR GATE.
                                                                  G2 00250
SLEVEL: GLEVEL(GI) TELLS WHICH LEVEL OF THE NETWORK GI IS IN.
                                                                  G2 00260
GSMALL: STORES INTERMEDIATE AND FINAL CALCULATED CSPF'S.
HLIST: HLIST(I, J) GIVES NAME OF I-TH GATE (OP EX. VAR.) IN NET-
                                                                  G2 00270
        WORK LEVEL J.
                                                                  G2 00280
  IDXO: LIST OF O-COORDINATES IN CSPFE OF THE GATE UNDER
                                                                  G2 00290
        CONSIDERATION.
                                                                  G2 00300
 IDXOE: LIST OF O-ERROR-COGRDINATES IN CSPFE OF THE GATE UNDER
                                                                  G2 00310
                                                                  G2 00320
        CONSIDERATION.
                                                                  G2 00330
  IDX1: LIST OF 1-COORDINATES IN CSPFE OF THE GATE UNDER
                                                                  G2 00340
        CONSIDERATION.
```

```
C
C
C
C
C
C
C
C
C
000
C
C
C
C
C
C
C
C
C
C
C
C
C
C
C
C
```

```
IDX1E: LIST OF 1-ERROR-COORDINATES IN CSPFE OF THE GATE UNDER
                                                                   G2 00350
                                                                   G2 00360
        CONSIDERATION.
 IFLAG: SAME AS EYEFLG IN SUBROUTINE PROCII.
                                                                   G2 00370
INC$MX: INC$MX(GI.GJ)>0 MEANS THERE EXISTS A CONNECTION FROM GATE G2 00380
        (OR EX. VAR.) GI TO GATE GJ. INC$MX(GI,GJ)=0 IF NOT.
                                                                   G2 00390
INPTCY: LISTS FOR EACH CORRESPONDING ENTRY OF F$1, HOW MANY INPUTSG2 00400
        HAVE A '1' IN THE POSITION INDICATED BY F$1.
                                                                   G2 00410
 IPATH: IPATH(GI)=1 MEANS GATE GI IS ON A PATH FROM A CERTAIN GATEG2 00420
        TO AN OUTPUT GATE. OTHERWISE IPATH(GI) = 0.
                                                                   G2 00430
 IPRED: IPRED(I,GJ) GIVES THE NAME OF THE I-TH GATE OR EX. VAR. ING2 00440
        A LIST OF GATES AND EX. VAR. FEEDING GJ.
                                                                   G2 00450
 ISUCC: ISUCC(I,GJ) GIVES THE NAME OF THE I-TH GATE FED BY GJ.
                                                                   G2 00460
        SAME AS JAYFLG IN SUBROUTINE PROCII.
                                                                   G2 00470
 JFLAG:
 KEYA: A FLAG INDICATING IF ANY ERROR COMPENSATION HAS BEEN
                                                                   G2 00480
        PERFORMED.
                                                                   G2 00490
  KEYB: A FLAG INDICATING IF ANY PRIMARY O-ERROR-COORDINATES HAS
                                                                   G2 00500
        BEEN COMPENSATED.
                                                                   G2 00510
 KFLAG: SAME AS KEIFLG IN PROCII.
                                                                   G2 00520
  LEVM: NUMBER OF LEVELS IN THE NETWORK (NOTE EX. VAR. ARE ALSO
                                                                   G2 00530
        ASSIGNED LEVELS JUST LIKE GATES).
                                                                   G2 00540
        LGLIST(J) TELLS NO. OF GATES AND EX. VAR. IN LEVEL J OF
                                                                   G2 00550
        NETWORK.
                                                                   G2 00560
   LIP: NUMBER OF PREDECESSORS FOR THE GATE UNDER CONSIDERATION.
                                                                   G2 C0570
LIPRED: LIPRED(GI) TELLS NO. OF IMMEDIATE PREDECESSORS OF GATE GI.G2 00580
LISTC: ORDERED LIST OF CONNECTIBLE INPUTS TO GCD. ORDERED BY
                                                                   G2 00590
        DECREASING NO. OF O'S IN GCO COVERED.
                                                                   G2 00600
 LISTL: ORDERED LIST OF GATES AND EX. VAR. WHICH ORIGINALLY FED
                                                                   G2 00610
        GCO AND WHICH HAVE NOT YET BEEN DISCONNECTED. ORDERED BY G2 00620
        DECREASING NO. OF ESSENTIAL 1'S.
                                                                   G2 00630
LISUCC: LISUCC(GI) TELLS NO. OF IMMEDIATE SUCCESSORS OF GATE (OR
                                                                   G2 00640
        EX. VAR.) GI.
                                                                   G2 00650
LMTS2: UPPER LIMIT OF THE NUMBER OF ELEMENTS IN SET S2.
                                                                   G2 00660
LPOTAB: FOR GATE GI, LPOTAB(GI) POINTS TO LAST ROW OF POTAB
                                                                   G2 00670
        CONCERNING GI.
                                                                   G2 00680
     M: NUMBER OF NETWORK CUTPUT GATES.
                                                                   G2 00690
     N: NUMBER OF EXTERNAL VARIABLES (OR INPUT FNC.) AVAILABLE.
                                                                   52 00700
NEPMAX: FOR ERROR COMPENSATION PROGRAMS. IF MORE THAN NEPMAX
                                                                   G2 00710
        ERROR POSITIONS OCCUR WHEN A PARTICULAR GATE IS REMOVED.
                                                                   G2 00720
        PROGRAM SKIPS ATTEMPT TO COMPENSATE FOR THAT GATE'S
                                                                   G2 00730
                  VALUE CAN BE SPECIFIED BY USER, OTHERWISE EQUAL G2 00740
        REMOVAL.
        TO ONE HALF OF N2 BY DEFAULT.
                                                                   G2 00750
   NM: SUM OF N PLUS M
                                                                   G2 00760
   NM1: SUM OF NM PLUS 1.
                                                                   G2 00770
   NN2: PRODUCT OF N AND N2.
                                                                   32 00780
                                                                   G2 00790
   NOS: NUMBER OF ELEMENTS IN SET S.
  NOS1: NUMBER OF ELEMENTS IN SET S1.
                                                                   G2 00800
NOS1SV: NUMBER OF ELEMENTS IN SET S1 BEFORE ENTERING SUBROUTINE
                                                                   G2 00810
        RPLCF.
                                                                   G2 00820
  NOS2: NUMBER OF ELEMENTS IN SET S2.
                                                                   G2 00830
  NOT1: NUMBER OF ELEMENTS IN SET T1.
                                                                   G2 00840
NOTISV: NUMBER OF ELEMENTS IN SET TI BEFORE ENTERING SUBROUTINE
                                                                   G2 00850
                                                                   G2 00860
        RPLCF.
   NOO: NUMBER OF ACTIVE ELEMENTS IN ARRAY ICXO.
                                                                   32 00870
  NOOE: NUMBER OF ACTIVE ELEMENTS IN ARRAY IDXOE.
                                                                   G2 00880
   NO1: NUMBER OF ACTIVE ELEMENTS IN ARRAY IDX1.
                                                                   G2 00890
  NOIE: NUMBER OF ACTIVE ELEMENTS IN ARRAY IDXIE.
                                                                   G2 00900
    NR: SUM OF N PLUS R.
                                                                   G2 00910
  NRN2: PRODUCT OF NR AND N2.
                                                                   G2 00920
 NRPLC: NRPLC(I) STORES THE NUMBER OF ELEMENTS IN RPLC(I,*)
                                                                   G2 00930
                                                    FOR I=1,2.
                                                                   G2 00940
    N1: SUM OF N PLUS 1.
                                                                   G2 00950
```

```
N2: NUMBER OF DIFFERENT INPUT COMBINATIONS TO BE CONSIDERED
              (USUALLY 2 TO THE POWER N).
      ORIGIN: ORIGIN(GI)=1 MEANS GI ORIGINALLY CONNECTED TO GCO.
              ORIGIN(GI)=0 MEANS GI DID NOT FEED GCO ORIGINALLY.
C
0
          P$: P$(1.-) CONSECUTIVELY LISTS CUTPUTS OF EVERY EX. VAR. AND G2 01000
C
              EVERY GATE (FOR EVERY INPUT COMBINATION): P$(1,1),...,
              P$(1, N2) FOR FIRST EX VAR; P$(1, N2+1), ..., P$(1, 2*N2) FOR
C
              SECOND EX VAR; ...; P$(1,N*N2+1),..., P$(1,N*N2+N2) FOR
              FIRST GATE; ETC. P$(2,-) IS USED AS WORK SPACE FOR
^
              CALCULATIONS ASSOCIATED WITH P$(1,-).
         PCO: FOR ERROR COMPENSATION PROCEDURES. PCO IS THE GATE
C
              REMOVED FROM ORIGINAL NETWORK TO OBTAIN CURRENT ALTERED
C
              NETWORK.
      POINTA: NOT USED.
C
      POINTC: POINTS TO LAST ELEMENT IN LISTC.
C
      POINTL: POINTS TO LAST ELEMENT IN LISTL.
      POINTR: POINTS TO LAST ELEMENT IN RNEC1 (IN SUBROUTINE SUBSTI).
      POTAB: POSSIBLE OUTPUT TABLE. HOLDS INFORMATION ABOUT ALL
              COMBINATIONS OF CONNECTIONS TO FORM NEW (AND HOPEFULLY
              USEFUL ) FUNCTIONS.
      PPOTAB: FOR GATE GI, PPOTAB(GI) POINTS TO FIRST OF A SEQUENCE OF
              ROWS OF POTAB CONCERNING GI.
C
           R: NUMBER OF GATES IN THE NETWORK (EXCLUDES EX VAR, ALSO
              NOTE SCME OF R GATES MAY BE ISOLATED).
        RPLC: RPLC(1,*) STORES THE SELECTED GATE'S IP GATES WHICH HAVE
                        ERROR-COORDINATES OF WEIGHT 2 OR ABOVE.
              RPLC(2,*) STORES THE SELECTED GATE'S IP GATES WHICH HAVE
                        AT LEAST ONE EPROR-COORDINATE OF WEIGHT 1.
      RSCONN: LIST OF CONNECTIONS ADDED TO A NETWORK (IN CODED FORM).
      RICCIN: LIST OF CONNECTIONS REMOVED FROM A NETWORK (CODED FORM).
           S: NO. OF CONNECTIONS ADDED TO A NETWORK. PCINTS TO LAST
              ENTRY IN RSCONN.
C
        SETS: SET S CONSISTING OF INPUTS OF THE GATE UNDER CONSIDERATIONG2 01280
C
              WHICH ARE TO BE REPLACED IF POSSIBLE.
              SET SI CONSISTING OF ELEMENTS OF SET S WHICH CAN BE
       SETS1:
              PEPLACED BY ELEMENTS IN SET S2.
C
       SETS2: SET S2 CONSISTING OF FUNCTIONS WHICH ARE CANDIDATES FOR
              REPLACING ELEMENTS IN SET S.
C
       SETT1: SET T1 CONSISTING OF ESSENTIAL ONES COVERED BY ELEMENTS ING2 01340
C
                                                                SET S1.
         STS: STARTING ELEMENT OF SET S.
      SUC$MX: SUC$MX(GI,GJ)>0 MEANS GATE GJ IS A SUCCESSOR OF GATE GI.
              SUC$MX(GI+GJ)=0 IF NOT.
        SUMP: SUM OF ALL ACTIVE INPUTS OF THE GATE UNDER CONSIDERATION. G2 01390
C
       SUMS2: SUM OF ALL ACTIVE ELEMENTS OF SET S2.
           T: NUMBER OF CONNECTIONS REMOVED FROM A NETWORK.
                                                              POINTS TO
              LAST ENTRY IN RTCONN.
        TIME: USED TO STORE AMOUNT OF ELAPSED COMPUTATION TIME.
       JNAME: MNEMONIC NAMES FOR EXTERNAL VARIABLES AND GATES.
      VF$UB1: POINTS TO LAST ELEMENT IN VF$1.
       VF$1: SIMILAR TO F$1, EXCEPT THIS LISTS JUST COMPONENT POSITIONS 2 01460
              (JF 0'S IN CSPF VECTOR OF GCO) COVERED ONLY BY REMAINING
C
              ORIGINALLY CONNECTED INPUTS TO GCO.
£,
      COMMON NEPMAX
     COMMON
               N
                                                             В
               D
                            , N2
                                                             NR
     1
                                              N1
               NM
                              KELAG
     2
                                                             COST
                                              JFLAG
```

NRN2

NM1

LEVM

G2 00960

G2 00970

G2 00980

G2 00990

G2 01010

G2 01020

G2 01030

G2 01040

G2 01050

G2 01060 G2 01070

G2 01080

G2 01090

G2 01100

G2 01110

G2 01120

G2 01130

G2 01140

G2 01150

G2 01160

G2 01170

G2 01180

G2 01190

G2 01200

G2 01210

G2 01220

G2 01230

G2 01240

G2 01250

G2 01260 G2 01270

G2 01290

G2 01300

G2 01310

62 01320 G2 01330

32 01350

G2 01360

G2 01370

G2 01380

G2 01400

G2 01410 G2 01420

32 01430

G2 01440

G2 01450

G2 01470

G2 01480

G2 01490 G2 01500 G2 01510

G2 01520

G2 01530

G2 01540

52 01550

G2 01560

NN2

```
ISUCC(40,40) , LISUCC(40)
                                               , IPRED(40,40) , LIPRED(40)
                                                                               G2 01570
      COMMON
                                                                . UNAME (40)
                INC$MX(40,40), SUC$MX(40,40), P$(2,1280)
                                                                               G2 01580
     1
                                               , HLIST (40,40)
                GLEVEL (40)
                              , LGLIST(40)
                                                               , TIME
                                                                               G2 01590
     2
                                               , 5
      COMMON
                               RTCONN(100)
                                                                • RSCCNN(100)
                                                                               G2 01600
      COMMON
                              , POINTA
                                               , ESS1S(40)
                                                                •F$1(32)
                                                                               G2 01610
                IFLAG
            ,F$UB1
                               ,INPTCV(32)
                                               ,LISTC(40)
                                                                , POINTC
                                                                               G2 01620
                                               ORIGIN(40)
     2
            ,LISTL(40)
                               , POINTL
                                                                , IPATH(40)
                                                                               G2 01630
            POINTR
                                               , VF $UB1
                              ,VF$1(32)
                                                                ,GSMALL(40,32)G2 01640
     3
                                               , LPOTAB(40)
      COMMON
              POTAB(200,42),PPOTAB(40)
                                                                , NRPLC(2)
                                                                               G2 01650
            • RPLC (2.40)
                              .IDX0(32)
                                               .IDX0E(32)
                                                                , IDX1 (32)
                                                                               G2 01660
     2
            . IDX1E(32)
                              .SUMP (32)
                                               .SETT1(32)
                                                                .NCT1
                                                                               G2 01670
            , SETS1(40)
                              , NOS1
                                               , SETS (40)
     3
                                                                · NOS
                                                                               G2 01680
                                               , SETS2(200)
                                                                               G2 01690
                               , SUMS 2(32)
                                                                . NOS2
     4
            , STS
                                               , KEYA
                                                                               G2 01700
     5
            .LIP
                               • NO0 E
                                                                .KEYB
            , NOO
                                               , NOIE
                                                                , $GT
                               .NC1
                                                                               G2 01710
     6
                              , SPW
                                               , $NOE
                                                                , GI
            , $LTH
                                                                               G2 01720
                              NOT1SV
                                              .NOS1SV
                                                               .LMTS2
                                                                               G2 01730
      COMMON
      DIMENSION CNTLIS(144), UGATE(40), UHEAD(20)
                                                                               G2 01740
                                     1/
      DATA KOUNTS /O/, UBLANK/ 1
                                                                               G2 01750
  990 READ(5,1000,END=500) UHEAD, N, M, R, A, B, UC, NEPMAX
                                                                               G2 01760
C
      NEPMAX IS THE MAXIMUM ALLOWABLE NUMBER OF ERRCR POSITIONS
                                                                               G2
                                                                                  01770
 1000 FORMAT (20A4/514, A4, I4)
                                                                               G2 01780
      KEYXC = 0
                                                                               G2 01790
      IF(UC.NE.UBLANK) KEYXC=1
                                                                               G2 01800
                                                                               G2 01810
      CALL PAGE
      CALL LINE(10)
                                                                               G2 01820
                                                                               G2
      KDUNT5=KOUNT5+1
                                                                                  01830
      PRINT 2, KOUNTS
                                                                               G2
                                                                                   01840
    2 FORMAT(20X, **** OPTIMAL NOR NETWORK ****, 50X, *PROBLEM NO. = *, I4 ) G2 01850
      CALL LINE(4)
                                                                               G2 01860
      PRINT 1005, UHEAD
                                                                               G2 01870
 1005 FORMAT (25X, 20A4)
                                                                               G2 01880
      CALL LINE(4)
                                                                               G 2
                                                                                  01890
      PRINT 10, N,M,A,B
                                                                               G2 01900
   10 FORMAT (30X, 'NUMBER OF VARIABLES = 1, 14 //
                                                                               G2 01910
     1
              30X, 'NUMBER OF FUNCTIONS =', I4 //
                                                                               G2 01920
     2
                                         = * , I 4//
              30X, COST COEFFICIENT A
                                                                               G2 01930
              47X,
                                     * B
                                         = 1.14)
                                                                               G2 01940
      CALL LINE(1)
                                                                               G2 01950
      IF(KEYXC.NE.O) GO TO 25
                                                                               G2 01960
      PRINT 21
                                                                               G2 01970
   21 FORMAT(1H0,29X,'--- UNCOMPLEMENTED VARIABLES
                                                                               G2 01980
      GO TO 30
                                                                               G2 01990
   25 CONTINUE
                                                                               G2 02000
      PRINT 28
                                                                               G2 02010
   28 FORMAT (1HO, 29X, '--- BOTH COMPLEMENTED AND UNCOMPLEMENTED VARIABLESG2 02020
     1 X, Y --- 1
                                                                               G2 02030
   30 CONTINUE
                                                                               G2 02040
      CALL LINE(5)
                                                                               G2 02050
C**** SET UP EXTERNAL VARIABLES ****
                                                                               G2 02060
      N2=2**N
                                                                               G2 02070
      IF (NEPMAX.EQ.O)NEPMAX = N2/2
                                                                               G2 02080
      H=N*N2
                                                                               G2 02090
      J=N2
                                                                               G2 02100
      L= 1
                                                                               G2 02110
      t = 0
                                                                               G2 02120
      D3 1011 II=1,N
                                                                               G2 02130
       J=J/2
                                                                               32 02140
       L=L*2
                                                                               G2 02150
       SN = 1
                                                                               G2 02160
       DO 1010 LL=1, L
                                                                               G2 02170
```

```
SN=-SN
                                                                                 G2 02180
         V = \{1 + SN\}/2
                                                                                 G2 02190
         DO 1009 JJ=1,J
                                                                                 G2 02200
          I = I + 1
                                                                                 G2 02210
          P$(1, I)=V
                                                                                 G2 02220
      IF (KEYXC.NE.O) P$ (1, I+H)=1-V
                                                                                 G2 02230
 1009
        CONTINUE
                                                                                 G2 02240
 1010 CENTINUE
                                                                                 G2 02250
 1011 CONTINUE
                                                                                 G2 02260
       IF(KEYXC.NE.O) N=N+N
                                                                                 G2 02270
      N1 = N+1
                                                                                 G2 02280
      NM=N+M
                                                                                 G2 02290
      NM1=NM+1
                                                                                 G2 02300
      NN2=N*N2+1
                                                                                 32 02310
      NR = N + R
                                                                                 G2 02320
      NRN2=NR*N2
                                                                                 G2 02330
      CALL OUTPUT(INC $MX, KEYXC)
                                                                                 G2 02340
C**** READ IN NETWORK INFORMATION AND SET UP INCSMX *****
                                                                                 G2 02350
      READ 1001.
                     CNTLIS
                                                                                 32 02360
 1001 FORMAT(1615)
                                                                                 G2 02370
      DO 1115 GI=1.NR
                                                                                 G2 02380
      DO 1115 GJ=1.NR
                                                                                 G2 02390
 1115 INCSMX(GI,GJ)=0
                                                                                 G2 02400
      DO 1120 I=1,144
                                                                                 G2 02410
       ITEM=CNTLIS(I)
                                                                                 32 02420
       IF(ITEM.EQ.O) GO TO 1119
                                                                                 G2 02430
       GI=ITEM/100
                                                                                 32 02440
       GJ=ITEM-100*GI
                                                                                 G2 02450
       INC $MX(GI,GJ)=1
                                                                                 G2 02460
                                                                                 G2 02470
       GO TO 1120
 1119 COST=A*R+B*(I-1)
                                                                                 G2 02480
       GO TO 1130
                                                                                 G2 02490
 1120 CONTINUE
                                                                                 G2 02500
 1130 CONTINUE
                                                                                 G2 02510
      CALL SUBNET
                                                                                 G2 02520
      CALL PVALUE
                                                                                 G2 02530
                                                                                 G2 02540
      CALL LINE(4)
      PRINT 1140, COST
                                                                                 G2 02550
 1140 FORMAT (20X, ORIGINAL NETWORK
                                          COST=', 15)
                                                                                 G2 02560
      CALL LINE(4)
                                                                                 G2 02570
      CALL TRUTH(P$.1)
                                                                                 G2 02580
      CALL LINE(4)
                                                                                 G2 02590
      CALL CKT(INC $MX, GLEVEL)
                                                                                 G2 02600
                                                                                 G2 02610
C**** ENTRY REDUNDANCY CHECK ****
                                                                                 G2 02620
      S = 0
                                                                                 G2 02630
      T = 0
                                                                                 G2 02640
      CALL UNNECE
                                                                                 G2 02650
      GATES = M
                                                                                 G2 02660
      C = 0
                                                                                 G2 02670
      DO 4 GI = 1.NR
                                                                                 G2 02680
      C = C + LISUCC(GI)
                                                                                 G2 02690
      IF(GI.LE.NM)GOTO4
                                                                                 S2 02700
      IF(LISUCC(GI).GT.O)GATES=GATES+1
                                                                                 G2 0271(
                                                                                 G2 02720
    4 CONTINUE
      DLDCST = A*GATES + B*(C)
                                                                                 32 02 73(
      T = 0
                                                                                 G2 0274(
      5=0
                                                                                 G2 02750
C
      INITIALIZE TIMER TO 10 MINUTES
                                                                                 G2 0276(
      CALL STIMEZ(60000)
                                                                                 G2 02770
                                                                                 G2 02780
      TIME = KTIMEZ(O)
```

```
PROCEDURE PROCIV
                                                                        G2 02790
门水水水水
      CALL PROCIV(3)
                                                                        G2 02800
      CALL FOR ELAPSED TIME
                                                                        G2 02810
      TIME = KTIMEZ(O) - TIME
                                                                        G2 02820
      CALL LINE(4)
                                                                        G2 02830
      PRINT 3915
                                                                        32 02840
 3916 FORMAT(20X, 'TIME ELAPSED =', 18, ' CENTISECONDS')
                                                                        G2 02850
 3915 FORMAT (20X, "NETWORK DERIVED BY PROCIV")
                                                                        G2 02860
                                                                        G2 02870
      PRINT 3916, TIME
                                                                        G2 02880
      CALL LINE(4)
     CALL TRUTH(P$,1)
                                                                        G2 02890
     CALL LINE(4)
                                                                        G2 02900
                                                                        G2 02910
     CALL CKT(INC $MX, GLEVEL)
      GATES = M
                                                                        32 02920
                                                                        G2 02930
     C = 0
                                                                        G2 02940
     DO 36 GI = 1.NR
      C = C + LISUCC(GI)
                                                                        32 02950
      IF(GI.LE.NM) GO TO 36
                                                                        G2 02960
      IF(LISUCC(GI).GT.O) GATES = GATES + 1
                                                                        G2 02970
                                                                        32 02980
  36 CONTINUE
      NEWCST = A*GATES + B*C
                                                                        G2 02990
      IF (NEWCST.LT. OLDCST) GO TO 37
                                                                        G2 03000
                                                                        32 03010
      PRINT 105
  105 FORMAT(1H ,10X, 'NO REDUNDANCY FOUND. ')
                                                                        G2 03020
                                                                        32 03030
      GO TO 990
                                                                        G2 03040
   37 CALL LINE(3)
      PRINT 320, NEWCST
                                                                        G2 03050
  320 FORMAT(9X.** A NETWORK DERIVED BY PROCIV*/9X.* COST=*.15.*.*)
                                                                        G2 03060
     GO TO 990
                                                                        32 03070
  500 STOP
                                                                        G2 03080
      END
                                                                        G2 03090
      SUBROUTINE PROCIV(VERSIN)
                                                                        G2 03100
      C
      IMPLICIT INTEGER*4(A-T, V-Z), REAL(U)
                                                                        G2 03120
                                                                        G2 03130
C
     DEFINITIONS OF "COMMON" VARIABLES CAN BE FOUND IN MAIN PROGRAM.
                                                                        G2 03140
C
                                                                        G2 03150
C
     VARIABLE DEFINITIONS:
                                                                        32 03160
C
         CA: NO. OF CONNECTIONS IN NETWORK AFTER CALLING MINI2.
                                                                        G2 03170
         CB: NO. OF CONNECTIONS IN NETWORK BEFORE CALLING MINI2.
                                                                        G2 03180
C
      CCNNS: NO. OF CONNECTIONS REMOVED BY MINI2.
                                                                        G2 03190
ſ,
     COUNTP: NUMBER OF CONSECUTIVE TIMES PROCII HAS BEEN CALLED WITHOUTG2 03200
C
             REMOVING A GATE.
                                                                        32 03210
C
         GA: NO. OF GATES IN NETWORK AFTER CALLING A SUBROUTINE.
                                                                        G2 03220
C
       GATE: GATE OF FOCUS FOR APPLICATION OF PROCEDURE (I.E., PROGRAM G2 03230
C
             CONCENTRATES ON ITS REMOVAL).
                                                                        G2 03240
C
      GATES: NO. OF GATES REMOVED BY MINIZ.
                                                                        G2 03250
C
         GB: NO. OF GATES IN NETWORK BEFORE CALLING A SUBROUTINE.
                                                                        G2 03260
C
      IMPROV: A PARAMETER RETURNED BY MINI2 INDICATING WHETHER IT WAS
                                                                        G2 03270
C
             SUCCESSFUL (=1) OR NOT (=0).
                                                                        G2 03280
Ç
     NEWCST: COST OF (POSSIBLY) NEW NETWORK DERIVED BY MINI2.
                                                                        G2 03290
C
     CNECNT: COUNTS NO. OF '1' COMPONENTS IN CSPF VECTOR OF SOME GATE. G2 03300
C
       ONES: ONES(GI) GIVES THE NUMBER OF '1' CCMPCNENTS IN GI'S CSPF
                                                                        G2 03310
C
             VECTOR
                                                                        G2 03320
C
     PORDER: AN ORDERING OF GATES (NO EX. VARS.) ACCORDING TO INCREAS- G2 03330
C
             ING NUMBERS OF 1'S IN THEIR CSPF VECTORS.
                                                                        G2 03340
C
     VERSIN: VERSION OF A PARTICULAR ORDERING TO BE USED BY PROCII IN
                                                                        G2 03350
C
             ASSIGNING COVERS.
                                                                        G2 03360
C
                                                                        G2 03370
```

```
I, J, MAX, MIN, PSUB ARE USED AS JUST TEMPORARY VARIABLES.
                                                                                G2 03380
C
C
                                                                                G2 03390
                                                                                G2 03400
C
      HOW TO INCREASE CAPACITY OF SUBROUTINE.
C
      DIMENSION: ONES(X), PORDER(X) - X EQUAL TO MAX NUMBER OF GATES
                                                                                G2 03410
C
                                         PLUS EXTERNAL VARIABLES.
                                                                                G2 03420
C
                                                                                G2 03430
      COMMON NEPMAX
                                                                                G2 03440
      COMMON
                N
                               * M
                                                  A
                                                                , B
                                                                                G2 03450
                                                .
                D
                               . N2
                                                , N1
                                                                  NR
                                                                                32 03460
            9
                                                , JFLAG
                               , KFLAG
                                                                , COST
                                                                                G2 03470
     2
                NM
            ,
                               , NRN2
                LEVM
                                                  NM1
                                                                  NN2
                                                                                G2 03480
                                                9
      COMMON
                ISUCC (40,40) , LISUCC (40)
                                               , IPRED(40,40) , LIPRED(40)
                                                                                G2 03490
                INC $MX(40,40), SUC $MX(40,40), P$(2,1280)
                                                                , UNAME (40)
                                                                                G2 03500
     1
                                              , HLIST(40,40) ,
                                                                  TIME
                GLEVEL(40)
                             , LGLIST(40)
                                                                                G2 03510
      COMMON
                               , RTCCNN(100)
                                                , S
                                                                , RSCONN(100) G2 03520
                               , POINTA
                                                ,ESS1S(40)
                IFLAG
                                                                •F$1(32)
      MCMMOD
                                                                                G2 03530
            ,F$UB1
                               ,INPTCV(32)
                                                ,LISTC(40)
                                                                , POINTC
                                                                                G2 03540
                               , POINTL
                                                                , IPATH(40)
                                                                                G2 03550
     2
            .LISTL(40)
                                                ORIGIN(40)
                               , VF$1(32)
            , POINTR
                                                .VF$UB1
                                                                .GSMALL(40,32)G2 03560
                                                , LPOTAB(40)
                                                                , NRPLC(2)
      COMMON
              POTAB(200,42),PPOTAB(40)
                                                                                G2 03570
                                                , IDX0E(32)
            ,RPLC(2,40)
                              ,IDX0(32)
                                                                , IDX1(32)
                                                                                G2 03580
     1
                               , SUMP ( 32)
            , IDX1E(32)
                                                ,SETT1(32)
                                                                ,NOT1
                                                                                G2 03590
     3
            , SETS1(40)
                               , NOS1
                                                ,SETS(40)
                                                                , NOS
                                                                                G2 03600
                                                                                G2 03610
            .STS
                               , SUMS 2 (32)
                                                , SETS2 (2001
                                                                NOS2
                                                , KEYA
     5
            .LIP
                               • NOOE
                                                                , KEYB
                                                                                G2 03620
            . NOO
                                                , NOIE
                               •N01
                                                                                G2 03630
     6
                                                                . $GT
                               , SPW
            , SLTH
                                                , $NOE
                                                                , GI
                                                                                G2 03640
      COMMON
                               NOTISV
                                               ,NOSISV
                                                               ,LMTS2
                                                                                G2 03650
C
                                                                                G2 03660
C
      "VERSIN" INDICATES THE ORDERING TO BE USED IN CALL TO PROCII
                                                                                G2 03670
C
        VERSION = 3 OR 5 RECOMMENDED, BUT VERSION = 1,2, OR 4 ALSO GOOD G2 03680
      DIMENSION ONES(40), PORDER(40)
                                                                                G2 03690
      GR = M
                                                                                32 03700
      CB = 0
                                                                                32 03710
      DO 2 I = 1.NR
                                                                                G2 03720
      CB = CB + LISUCC(I)
                                                                                32 03730
      IF(I.LE.NM) GO TO 2
                                                                                G2 03740
      IF(LISUCC(I).GT.0) GB = GB + 1
                                                                                G2 03750
                                                                                G2 03760
    2 CONTINUE
    1 CALL MINIZ(IMPROV)
                                                                                G2 03770
                                                                                G2 03780
      IF (IMPROV.NE.O) GD TO 1
      GA = M
                                                                                G2 03790
      CA = 0
                                                                                G2 03800
      DD 3 I = 1, NR
                                                                                G2 03810
      CA = CA + LISUCC(I)
                                                                                G2 03820
                                                                                G2 03830
      IF(I.LE.NM) GC TO 3
      IF(LISUCC(I).GT.O) GA = GA + 1
                                                                                G2 03840
                                                                                G2 03850
    3 CCNTINUE
      GATES = GB - GA
                                                                                G2 03860
      CCNNS = CB - CA
                                                                                G2 03870
      PRINT 1000, GATES, CONNS
                                                                                G2 03880
 1000 FORMAT(5x,16, GATES AND',16, CONNECTIONS HAVE BEEN REMOVED BY MIG2 038)0
                                                                                32 03900
     1912 OURING INITIALIZATION FOR PROCIVE)
      IF(CONNS . EQ.O) GO TO 4
                                                                                G2 03910
      CALL LINE(4)
                                                                                G2 03920
      PRINT 1001
                                                                                G2 03930
                                                                                G2 03940
 1001 FORMAT(20X, 'NETWORK DERIVED BY MINI2 AS PART OF PROCIV')
                                                                                G2 03950
      CALL LINE(4)
      CALL TRUTH(P$,1)
                                                                                52 03960
                                                                                G2 03970
      CALL LINE (4)
      CALL CKT(INC & MX, GLEVEL)
                                                                                G2 03980
```

```
32 03990
      NEWCST = A*GA + B*CA
                                                                             G2 04000
      PRINT 1002 NEWCST
 1002 FORMAT(9X, ** A NETWORK DERIVED BY MINI2 STEP OF PROCIV'/9X, COST G2 04010
                                                                             G2 04020
     1=1, [6,1.1)
      COUNT THE NUMBER OF 1'S IN THE CSPF VECTOR FOR EACH GATE
                                                                             G2 04030
C
                                                                             G2 04040
    4 MAX = 0
      DO 5 I = N1,NR
                                                                             G2 04050
      ONECNT = 0
                                                                             G2 04060
      DO 6 J = 1.N2
                                                                             G2 04070
      IF (GSMALL(I, J). LE. 0)GO TO 6
                                                                             G2 04080
      ONECNT = ONECNT + 1
                                                                             G2 04090
                                                                             G2 04100
    6 CONTINUE
                                                                             G2 04110
      IF (ONECNT. GT. MAX) MAX=ONECNT
      CNES(I) = ONECNT
                                                                             G2 04120
                                                                             G2 04130
    5 CONTINUE
                                                                             G2 04140
      MAX = MAX + 1
      MIN = -1
                                                                             G2 04150
      PSUB = 1
                                                                             G2 04160
                                                                             G2 04170
    7 MIN = MIN + 1
      IF (MIN. EQ. MAX) GO TO 8
                                                                             G2 04180
                                                                             G2 04190
      DO 9 I = N1,NR
      IF (ONES (I) . NE . MIN) GO TO 9
                                                                            G2 04200
      PORDER(PSUB) = I
                                                                            G2 04210
                                                                             G2 04220
      PSUB = PSUB + 1
    9 CONTINUE
                                                                             G2 04230
                                                                             G2 04240
      GOTO7
    8 CONTINUE
                                                                             G2 04250
                                                                             G2 04260
      I = 1
                                                                            G2 04270
      COUNTR = 0
      CALCULATE GB, THE NUMBER OF GATES BEFORE CALLING PROCII
                                                                            G2 04280
      GS = M
                                                                            G2 04290
                                                                            G2 04300
      DO 10 J = 1.NR
                                                                             G2 04310
      IF(J.LE.NM) GO TO 10
      IF(LISUCC(J).GT.O) GB = GB + 1
                                                                             G2 04320
   10 CONTINUE
                                                                             G2 04330
   11 IF(I.GT.R)I = 1
                                                                             G2 04340
      GATE = PORDER(I)
                                                                             G2 04350
      IF(GATE.LE.NM) GO TO 12
                                                                             G2 04360
                                                                            G2 04370
      CALL PROCII(VERSIN, 2, GATE, 1)
C
      CALCULATE GA, THE NUMBER OF GATES AFTER CALLING PROCII
                                                                             G2 04380
                                                                            G2 04390
      GA = M
      DO 13 J = 1.NR
                                                                             G2 04400
      IF(J.LE.NM)GO TO 13
                                                                             G2 04410
                                                                            G2 04420
      IF(LISUCC(J).GT.O) GA = GA + 1
   13 CONTINUE
                                                                             G2 04430
                                                                             G2 04440
      IF(GA. EQ.GB) GO TO 12
      COUNTR = 0
                                                                             G2 04450
                                                                            G2 04460
      I = I + 1
      GB = GA
                                                                             G2 04470
      GO TO 11
                                                                             G2 04480
   12 COUNTR = COUNTR + 1
                                                                             G2 04490
                                                                            G2 04500
      I = I + 1
      IF(COUNTR.GE.R) RETURN
                                                                             G2 04510
      GO TO 11
                                                                            G2 04520
```

G2 04530

END

IBLIOGRAPHIC DATA 1. Report No. UIUCDCS-R-75-698	3. Recipient's Accession No.	
Title and Subtitle PROGRAM MANUAL: NOR NETWORK TRANSDUCTION BASED ON CONNECT-	5. Report Date February 1975 6.	
	8. Performing Organization Rept. No.	
Performing Organization Name and Address Department of Computer Science University of Illinois at Urbana-Champaign Urbana, Illinois 61801	10. Project/Task/Work Unit No.  11. Contract/Grant No.  NSF GJ-40221	
2 Sponsoring Organization Name and Address National Science Foundation 1800 G Street, N.W. Washington, D.C. 20550	13. Type of Report & Period Covered Technical 14.	

5. Supplementary Notes

This paper explains the operation and usage of two FORTRAN computer programs, NETTRA-Gl and NETTRA-G2, developed for NOR-network transduction (transformation and

reduction).

Existing (non-optimal) NOR-gate networks and their required output functions are given to the programs as input. The programs, in general, add, change, and/or delete connections in the network in an effort to reduce the cost of the network (defined in terms of numbers of gates and connections) as much as possible. Gates are examined individually; their input connections and potential input connections are evaluated under certain conditions of connectability and disconnectability in order to effect the changes in network configuration and thus reduce network cost.

These programs are only two out of a whole system of programs, designated by the name "NETTRA" (for NETwork TRAnsduction), which implement different NOR-network transduction procedures.

7. Key Words and Document Analysis. 17a. Descriptors

Logic design, logic circuits, logical elements, programs (computers).

7b. Identifiers/Open-Ended Terms

Computer-aided-design, permissible functions, network transduction, network transformation, redundant networks, near-optimal networks, NOR, NAND, CSPF, program manual, NETTRA-G1, NETTRA-G2.

7c. COSATI Field/Group

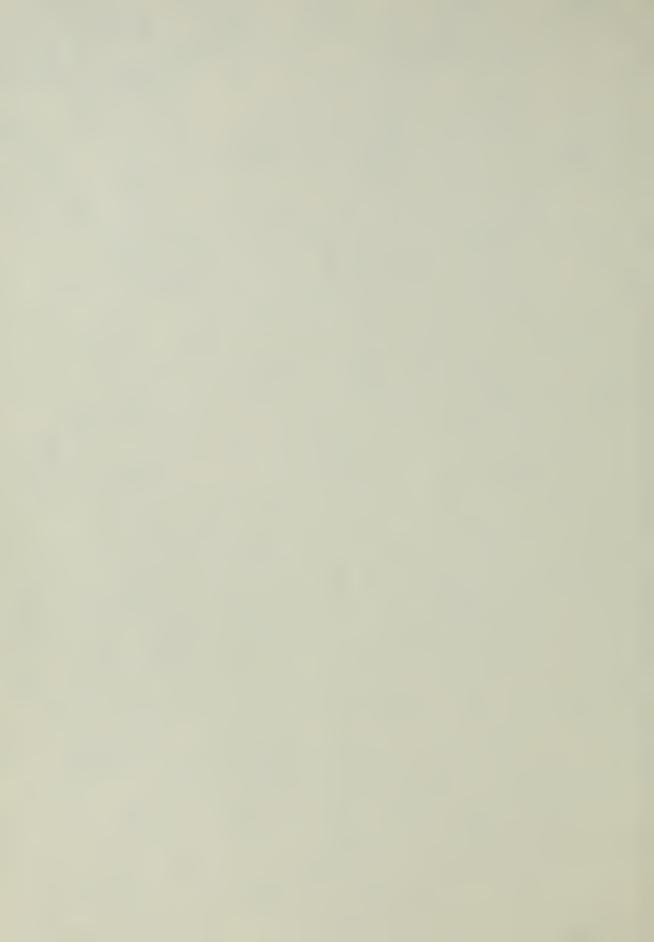
3. Availability Statement Release unlimited	19. Security Class (This Report) UNCLASSIFIED 20. Security Class (This	21. No. of Pages
DRM NTIS-35 (10-70)	Page UNCLASSIFIED	USCOMM-DC 40329-P71

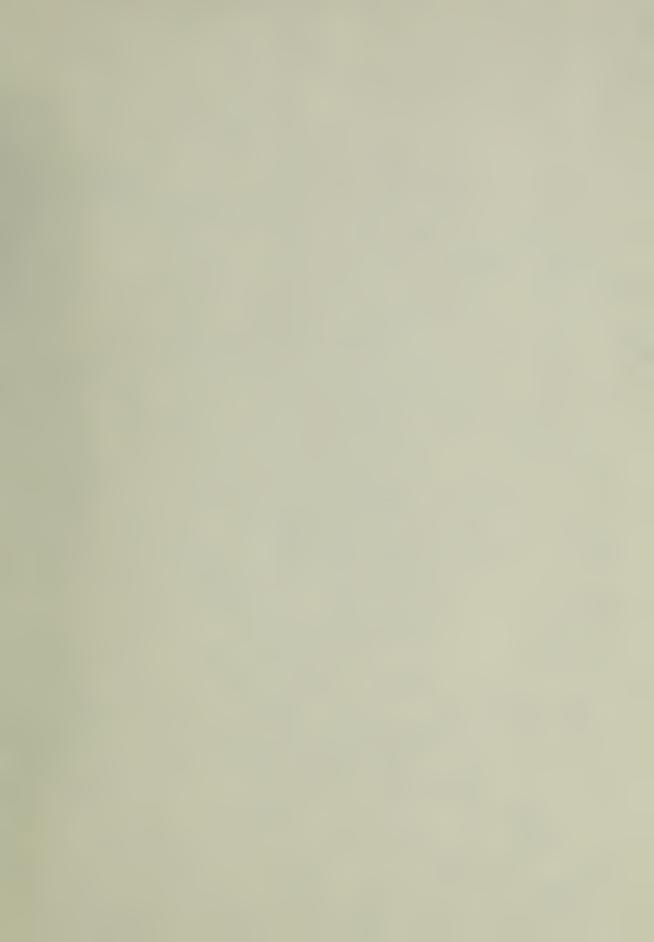














12 m = 1 1377



UNIVERSITY OF ILLINOIS-URBANA
510.84 IL6R no. C002 no.698-702(1975
Report /
3 0112 088401747